NOT MEASUREMENT SENSITIVE

NOTE: The cover page of this standard has been changed for administrative reasons. There are no other changes to this document.

MIL-STD-2500A
12 October 1994
SUPERSEDING
MIL-STD-2500
18 June 1993

DEPARTMENT OF DEFENSE
INTERFACE STANDARD

NATIONAL IMAGERY TRANSMISSION
FORMAT
(VERSION 2.0)

FOR THE
NATIONAL IMAGERY TRANSMISSION FORMAT STANDARD

AMSC N/A

AREA INST

DISTRIBUTION STATEMENT A. Approved for public release; distribution is unlimited.

FOREWORD

1. The National Imagery Transmission Format Standard (NITFS) is the standard for formatting digital imagery and imagery-related products and exchanging them among members of the Intelligence Community (IC) as defined by the Executive Order 12333, the Department of Defense (DOD), and other departments and agencies of the United States Government, as governed by Memoranda of Agreement (MOA) with those departments and agencies.

2. The National Imagery Transmission Format Standard Technical Board (NTB) developed this standard based upon currently available technical information.

3. The DOD and members of the Intelligence Community are committed to interoperability of systems used for formatting, transmitting, receiving, and processing imagery and imagery-related information. This standard describes the National Imagery Transmission Format (NITF) file format and establishes its application within the NITFS.

4. Beneficial comments (recommendations, additions, deletions) and any pertinent data which may be of use in improving this document should be addressed to Defense Information Systems Agency (DISA), Joint Interoperability and Engineering Organization (JIEO), Center for Standards (CFS), Attn: TBCE, 10701 Parkridge Blvd., Reston, VA 22091-4398 by using the Standardization Document Improvement Proposal (DD Form 1426) appearing at the end of this document or by letter.

CONTENTS

CONTENTS

PARAGRAPH PAGE

CONTENTS

PARAGRAPH PAGE

FIGURE

TABLE

CONTENTS

PARAGRAPH PAGE

APPENDIX

1. SCOPE

1.1 Scope. This standard establishes the requirements for the file format component of the National Imagery Transmission Format Standard (NITFS). The file format described in this document is called the National Imagery Transmission Format (NITF). The NITFS is a collection of related standards and specifications developed to provide a foundation for interoperability in the dissemination of imagery and imagery-related products among different computer systems. An overview of the component documents of the NITFS can be found in MIL-HDBK-1300A.

1.2 Content. This standard provides a detailed description of the overall structure of the file format, as well as specification of the valid data content and format for all fields defined within a NITF file. Several NITF implementation issues are addressed in the appendix. Issues pertinent to the use of NITF as the message format for imagery transmission are described in the transmission protocol component of NITFS, MIL-STD-2045-44500. An example of NITF as the basis for message formation in tactical communications is provided in section 6.

1.3 Applicability. This standard is applicable to the Intelligence Community and the DOD. It is mandatory for all Secondary Imagery Dissemination Systems (SIDS) in accordance with the memorandum by the Assistant Secretary of Defense for Command, Control, Communications and Intelligence ASD(C^3I), Subject: National Imagery Transmission Format Standard (NITFS), 12 August 1991. This directive shall be implemented in accordance with the JIEO Circular 9008, and MIL-HDBK-1300A. New equipment and systems, those undergoing major modification, or those capable of rehabilitation, shall conform to this standard.

1.4 Tailoring task, method, or requirements specification. Certifiable implementation of the NITF for support of interoperability is subject to constraints not specified in this standard. Pertinent compliance requirements are defined in JIEO Circular 9008.

1.5 Relationship to earlier formats. NITF Version 2.0 is an extension of the NITF Version 1.1. It includes the addition of (1) symbol graphics encoded using Computer Graphics Metafile (CGM), (2) image compression using Joint Photographic Experts Group Image Compression (JPEG), (3) extension of Adaptive Recursive Interpolated Differential Pulse Code Modulation (ARIDPCM) including 11-bit pixels and synchronization codes, (4) Bi-Level Image Compression, and (5) Vector Quantization image compression. NITF Version 2.0 also expands the range of certain header fields and modifies the way some fields are interpreted. Therefore, software designed to read both NITF Version 1.1 and Version 2.0 files needs to treat these fields differently. For this document, NITF refers to NITF Version 2.0.

2. APPLICABLE DOCUMENTS

2.1 Government documents.

2.1.1 Specifications, standards and handbooks. The following specifications, standards, and handbooks form a part of this document to the extent specified herein. Unless otherwise specified, the issue of these documents are those listed in the issue of the Department of Defense Index of Specifications and Standards (DODISS) and supplement thereto, cited in the solicitation.

STANDARDS

FEDERAL

FED-STD-1037B	-	Telecommunications: Glossary of Telecommunication Terms.

FEDERAL INFORMATION PROCESSING STANDARDS

FIPS PUB 10-3	-	Countries, Dependencies, Areas of Special Sovereignty, and Their Principal Administrative Divisions.
FIPS PUB 128	-	Computer Graphics Metafile (CGM) [adaptation of American National Standards Institute (ANSI) X3.122-1986.

MILITARY

MIL-STD-2301	-	Computer Graphics Metafile (CGM) for the National Imagery Transmission Format Standard (NITFS).
MIL-STD-2045-44500	-	Tactical Communications Protocol 2 (TACO2) for the National Imagery Transmission Format Standard (NITFS).
MIL-STD-188-196	-	Bi-Level Image Compression for the National Imagery Transmission Format Standard (NITFS).
MIL-STD-188-197A	-	Adaptive Recursive Interpolated Differential Pulse Code Modulation Image Compression (ARIDPCM) for the National Imagery Transmission Format Standard (NITFS).
MIL-STD-188-198A	-	Joint Photographic Experts Group (JPEG) Image Compression for the National Imagery Transmission Format Standard (NITFS).
MIL-STD-188-199	-	Vector Quantization Decompression for the National Imagery Transmission Format Standard (NITFS).

HANDBOOK

MILITARY

MIL-HDBK-1300A	-	National Imagery Transmission Format Standard (NITFS) Handbook.

(Unless otherwise indicated, copies of federal and military specifications, standards, and handbooks are available from the Standardization Documents Order Desk, 700 Robbins Avenue, Building #4, Section D, Philadelphia, PA 19111-5094.)

(Copies of Federal Information Processing Standards (FIPS) are available to Department of Defense activities from the Standardization Documents Order Desk, 700 Robbins Avenue, Building #4, Section D, Philadelphia, PA 19111-5094. Others must request copies of FIPS from the National Technical Information Service, 5285 Port Royal Road, Springfield, VA 22161-2171.)

2.1.2 Other Government documents, drawings, and publications. The following other Government documents, drawings, and publications form a part of this document to the extent specified. Unless otherwise specified, the issues are those cited in the solicitation.

JCS Pub 6-04	-	United States Message Text Format.
Technical Manual No. 5-241-1	-	GRIDS and GRID REFERENCES, Department of the Army.
DISA/JIEO Circular 9008	-	NITFS Certification Test and Evaluation Program Plan.

(Copies of JCS Pub 6-04 and Technical Manual No. 5-241-1 may be obtained from the appropriate Military Service Publication Center through the Military Service assigned administrative support. Copies of DISA/JIEO Circular 9008 may be obtained from DISA/JITC ATTN: GADB Bldg. 57305 Fort Huachuca, AZ 85613-7020.)

2.2 Non-Government publications. The following document(s) form a part of this document to the extent specified herein. Unless otherwise specified, the issues of the documents which are DOD adopted are those listed in the issue of the DODISS cited in the solicitation. Unless otherwise specified, the issues of documents not listed in the DODISS are the issues of the documents cited in the solicitation.

AMERICAN NATIONAL STANDARDS INSTITUTE (ANSI)

ANSI X3.4 - 1986 - American National Standard Code for Information Interchange (ASCII).

(Non-Government standards and publications are usually available from the organizations that prepare or distribute the documents. These documents also may be available in or through libraries or other informational services.)

2.3 Order of precedence. In the event of a conflict between the text of this document and the references cited herein, the text of this document takes precedence. Nothing in this document, however, supersedes applicable laws and regulations unless a specific exemption has been obtained.

3. DEFINITIONS

3.1 Acronyms used in this standard. The following definitions are applicable for the purpose of this standard. In addition, terms used in this standard and defined in the FED-STD-1037B shall use the FED-STD-1037B definition unless noted.

a. AL - Attachment Level

b. ANSI - American National Standards Institute

c. ARIDPCM - Adaptive Recursive Interpolated Differential Pulse Code Modulation

d. ASCII - American Standard Code for Information Interchange

e. ASD(C^3I) - Assistant Secretary of Defense for Command, Control, Communications, and Intelligence

f. CCIR - International Radio Consultative Committee

g. CCITT - International Telegraph and Telephone Consultative Committee (Organized under the auspices of International Telecommunications Union (ITU))

h. CFS - Center for Standards

i. CGM - Computer Graphics Metafile

j. CRT - Cathode Ray Tube

k. C^3I - Command, Control, Communications, and Intelligence

l. DES - Data Extension Segments

m. DISA - Defense Information Systems Agency

n. DL - Display Level

o. DOD - Department of Defense

p. DPCM - Differential Pulse Code Modulation

q. FIPS - Federal Information Processing Standard

r.	IC	-	(1) Intelligence Community (2) Image Compression
s.	ITU	-	International Telecommunications Union
t.	JIEO	-	Joint Interoperability and Engineering Organization (formerly JTC^3A)
u.	JPEG	-	Joint Photographic Experts Group
v.	LSB	-	Least Significant Bit
w.	LUT	-	Look-Up Table
x.	MOA	-	Memoranda of Agreement
y.	MSB	-	Most Significant Bit
z.	MTF	-	Message Text Format
aa.	NITF	-	National Imagery Transmission Format
ab.	NITFS	-	National Imagery Transmission Format Standard
ac.	NPPBH	-	Number of Pixels Per Block Horizontal
ad.	NPPBV	-	Number of Pixels Per Block Vertical
ae.	NTB	-	National Imagery Transmission Format Standard Technical Board
af.	OADR	-	Originating Agency's Determination is Required
ag.	RGB	-	Red, Green, Blue
ah.	SID	-	Secondary Imagery Dissemination
ai.	SIDS	-	Secondary Imagery Dissemination System
aj.	TACO2	-	Tactical Communications Protocol 2
ak.	UN	-	United Nations
al.	USMTF	-	United States Message Text Format (formerly JINTACCS)

am.	VQ	-	Vector Quantization
an.	YCbCr601	-	Y = Brightness of signal, Cb = Chrominance (blue), Cr = Chrominance (red)

3.2 Definition of terms. The definitions used in this document are defined as follows:

a. Alphanumeric - For the purpose of MIL-STD-2500A, fields that may contain any printable ASCII characters (including punctuation marks) are indicated as "Alphanumeric" in the Value Range specification. The reader is warned that this is a nonstandard use of the term. The allowable range of values for numeric fields typically is indicated in the form N-M, where N and M are the minimum and maximum values, respectively.

b. American Standard Code for Information Interchange (ASCII) - The standard code, using a coded character set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange among data processing systems, data communications systems, and associated equipment.

c. Bandwidth - 1. The difference between the limiting frequencies within which performance of a device, in respect to some characteristic, falls within specified limits. 2. The difference between the limiting frequencies of a continuous frequency band.

d. Block - For the purpose of MIL-STD-2500A, a block is a rectangular array of pixels. An image consists of the union of one or more non-overlapping blocks. (Synonymous with tile.)

e. Blocked Image Mask - A structure which identifies the blocks in a blocked (tiled) image which contain no valid data, and which are not transmitted or recorded. The structure allows the receiver to recognize the offset for each recorded/transmitted block. For example, a 2 x 2 blocked image which contained no valid data in the second block (block 1) would be recorded in the order: block 0, block 2, block 3. The blocked image mask would identify block 1 as a non-recorded/non-transmitted block, and would allow the receiving application to construct the image in the correct order.

f. Briefing board - A briefing aid that includes an exploited, annotated hardcopy image and other textual and/or graphical material that presents significant intelligence information.

g. Brightness - An attribute of visual perception, in accordance with which a source appears to emit more or less light. Note 1: Usage should be restricted to non-quantitative reference to physiological sensations and perceptions of light. Note 2: "Brightness" was formerly used as a synonym for the photometric term "luminance" and (incorrectly) for the radiometric term "radiance." For the purpose of NITFS, larger pixel values represent higher intensity, and lower pixel values represent lower intensity levels.

h. Broadcast operation - The transmission of information so that it may be simultaneously received by stations that usually make no acknowledgement.

i. Byte - A sequence of N adjacent binary digits, usually treated as a unit, where N is a non zero integral number. Note: In pre-1970 literature, "byte" referred to a variable length field. Since that time the usage has changed so that now it almost always refers to an eight-bit field. This usage predominates in computer and data transmission literature; when so used, the term is synonymous with "octet." For the purpose of MIL-STD-188-198A (JPEG), a byte is defined as an eight-bit octet.

j. Character - 1. A letter, digit, or other symbol that is used as part of the organization, control, or representation of data. 2. One of the units of an alphabet. Note: For MIL-STD-2301, a character (ANSI 3.4-1986 7-bit ASCII code padded into 8-bits) is an unsigned integer between and including 32 and 126 and is specified in this document using the character array C1, C2, ... Cn.

k. Conditional - In the context of NITF, a data field whose existence depends on the value used in a previous field.

l. Data - Representation of facts, concepts, or instructions in a formalized manner suitable for communication, interpretation, or processing by humans or by automatic means. Any representations such as characters or analog quantities to which meaning is or might be assigned.

m. Data communication - The transfer of information between functional units by means of data transmission according to a protocol.

n. Differential Pulse-Code Modulation (DPCM) - A version of pulse-code modulation in which an analog signal is sampled, and the difference between the actual value of each sample and its predicted value (derived from the previous sample or samples) is quantized and is converted by encoding to a digital signal. Note: There are several variations on differential pulse-code modulation.

o. Effectivity - Some of the capabilities specified in this document are not required as of the issue date of the document. All such capabilities are marked with effectivity numbers, (for example, Effectivity 1). Each effectivity number will be replaced by a specific date in subsequent releases of this document.

p. Gray scale - An optical pattern consisting of discrete steps or shades of gray between black and white.

q. Interface - 1. A concept involving the definition of the interconnection between two equipment items or systems. The definition includes the type, quantity, and function of the interconnecting circuits and the type, form, and content of signals to be interchanged via those circuits. Mechanical details of plugs, sockets, and pin numbers, etc., may be included within the context of the definition. 2. A shared boundary, e.g., the boundary between two subsystems or two devices. 3. A boundary or point common to two or more similar or dissimilar command and control systems, subsystems, or other entities against which or at which necessary information flow takes place. 4. A boundary or point common to two or more systems or other entities across which useful information flow takes place. (It is implied that useful information flow requires the definition of the interconnection of the systems which enables them to interoperate.) 5. The process of interrelating two or more dissimilar circuits or systems 6. The point of interconnection between user terminal equipment and commercial communication-service facilities.

r. International Telecommunication Union (ITU) - A civil international organization established to promote standardized telecommunication on a worldwide basis. Note: The CCIR and CCITT are committees under the ITU. The ITU headquarters is located in Geneva, Switzerland. While older than the United Nations (UN), it is recognized by the UN as the specialized agency for telecommunications.

s. Look-Up Table - A table where each data value of a pixel corresponds to an entry in the table.

t. Network - 1. An interconnection of three or more communicating entities and (usually) one or more nodes. 2. A combination of passive or active electronic components that serves a given purpose.

u. Optional - In the context of NITF, a data field that must be present, but may not have valid data.

v. Parity - In binary-coded systems, the oddness or evenness of the number of ones in a finite binary stream. Note: By the addition of one extra bit, a bit stream can be forced to a specified parity state. This is often used as a simple error-detection check and will detect (but not correct) the occurrences of any single bit error in the field.

w. Printable American Standard Code for Information Interchange - A subset of the ASCII code that causes a character to be printed.

x. Protocol - 1. [In general], A set of semantic and syntactic rules that determines the behavior of functional units in achieving communication. Note: Protocols may govern portions of a network, types of service, or administrative procedures. For example, a data link protocol is the specification of methods whereby data communication over a data link is performed in terms of the particular transmission mode, control procedures, and recovery procedures. 2. In layered communication system architecture, a formal set of procedures that are adopted to facilitate functional interoperation within the layered hierarchy.

y. Pseudocolor - A user-defined mapping of N bits into arbitrary colors.

z. Required - In the NITF context, a data field that must be present and filled with valid data.

aa. Resolution - 1. The minimum difference between two discrete values that can be distinguished by a measuring device. Note: High resolution does not necessarily imply high accuracy. 2. The degree of precision to which a quantity can be measured or determined. 3. A measurement of the smallest detail that can be distinguished by a sensor system under specific conditions.

ab. Secondary Imagery Dissemination (SID) - The process of post-collection electronic dissemination of Command, Control, Communications, and Intelligence (C^3I) digital imagery and associated data, over a time interval ranging from near-real-time to a period of days, at a level of quality determined by receiver requirements.

ac. Secondary Imagery Dissemination System (SIDS) - The equipment and procedures used in the electronic transmission and receipt of exploited non-original quality imagery and imagery products in other than real or near-real time.

ad. Tile - (See block.)

ae. Transparent Pixel - A fill pixel within an image block. Transparent pixels are recorded/transmitted to ensure that each block is filled with contiguous pixel values, but should be interpreted as having no meaning.

af. Transparent Pixel Mask - A data structure which identifies recorded/transmitted image blocks which contain transparent pixels. The transparent pixel mask allows the application to easily identify blocks which require special interpretation due to transparent pixel content.

ag. Vector Quantization - A compression technique in which many groups of pixels in an image are replaced by a smaller number of image codes. A clustering technique is used to develop a codebook of "best fit" pixel groups to be represented by the codes. Compression is achieved because the image codes can be recorded using fewer bits than the original groups of pixels they represent.

ah. Zulu - Synonym - Coordinated Universal Time. Formerly a synonym for Greenwich Mean Time.

4. GENERAL REQUIREMENTS

4.1 Background. The DOD and the IC use multiple types of systems for the reception, transmission, storage, and processing of images, text, and associated data. Without special efforts, the file format used in systems of one service or agency are likely to be incompatible with formats used in different systems of another organization. Incompatibilities also may exist among different systems within the same organization. Since each system may use a unique, internal data representation, a common format for exchange of information across systems is needed for interoperability of systems within and across service organizations and agencies. As the need for imagery-related systems grows, their diversity is anticipated to increase. Also anticipated is that the need to exchange data between organizations will increase, even though systems of each organization must retain their own individual characteristics and capabilities. This document defines the NITF, the standard file format for digital imagery and imagery-related products to be used by the DOD and the IC. The NITF provides a common basis for storage and digital interchange of images and associated data among existing and future systems. The NITF can be used to support interoperability by simultaneously providing a data format for shared access applications, while also serving as a standard message format for dissemination of images and associated data (text, symbols, labels) via digital communications. For historical reasons, the latter role as a transmission format gives NITF its name.

4.2 Relationship of NITF to the NITFS operations concept. In the NITFS concept, data interchange between systems is enabled by a cross-translation process. Each system will have processors to translate between the system's internal representation for files and data and the NITF file format. A system from which data is to be transferred is envisioned to have a translation module that accepts information, structured according to the system's internal representation for images, text files, and other data, and assembles this information into one file in the standard NITF file format. Then the file will be transmitted to one or more recipients as a message using a user-chosen communication protocol. NITFS specifies TACO2 for use in tactical communications. The receiving systems will reformat the message, converting it into one or more files structured as required by the internal representation of the receiving station. The functional architecture of this cross-translation process is shown on figure 1. In the diagram, the terms "Native_1 Files" and "Native_2 Files" refer to files represented in a way potentially unique to the sending (system 1) or receiving (system 2) system, respectively. Using the NITF, each system must be compliant with only one external format that will be used for communication with all other participating systems. The standard format allows a system to send data to several other systems with one transmission since each receiving station converts the message into its own internally acceptable form. Because each receiving station can translate selectively and permanently store only those portions of data in the received file that are of interest, a station may transmit all of its data in one message containing the one file, even though some of the target systems may be unable to process certain elements of the data usefully.

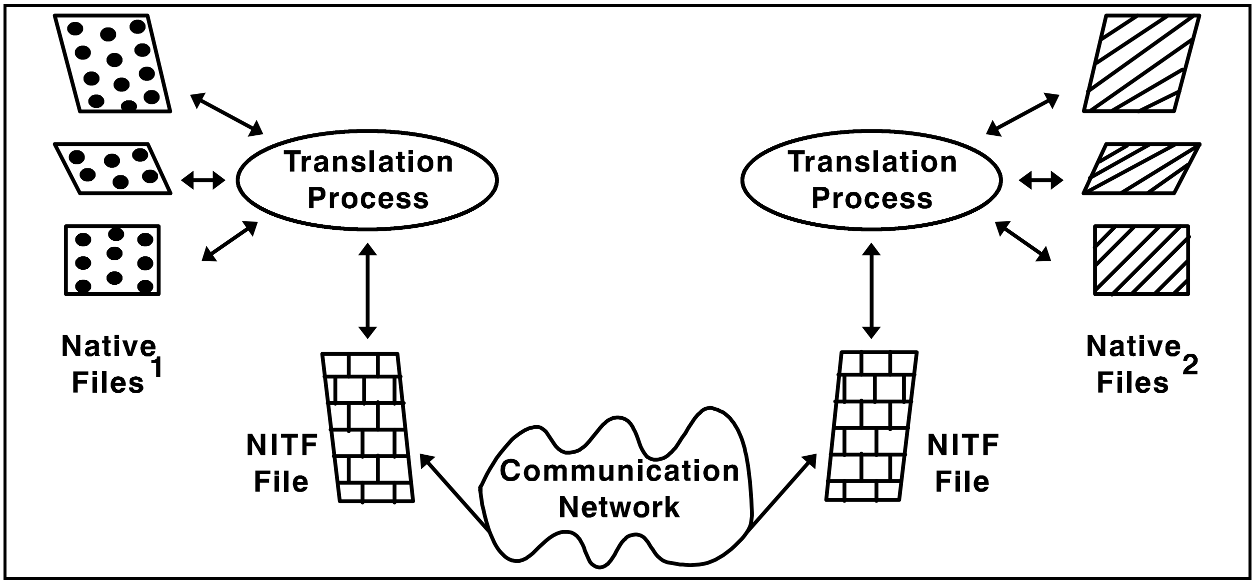


FIGURE 1. NITFS functional architecture.

4.3 NITF design objectives. The design objectives of the NITF are as follows:

a. To provide a means whereby diverse systems can share imagery and associated data.

b. To allow a system to send comprehensive information within one message to users with diverse needs or capabilities, allowing each user to select only those data items that correspond to their needs and capabilities.

c. To minimize the cost and schedule required to achieve such capability.

4.4 NITF characteristics. To serve a varied group of users communicating multiple types of imagery and imagery-related data who are using differing hardware and software systems, the NITF strives to possess the following characteristics:

a. Completeness - allows transmission of all needed imagery and imagery-related data.

b. Simplicity - requires minimal preprocessing and postprocessing of transmitted data.

c. Minimal overhead - minimizes formatting overhead, particularly for those users transmitting only a small amount of data and for bandwidth-limited users.

d. Universality - provides universal features and functions without requiring commonality of hardware or software.

4.5 NITF general requirements. The NITF is specified to satisfy several general requirements in response to the role it plays in the NITFS functional architecture. These requirements are:

a. To be comprehensive in the kinds of data permitted in the file within the image-related objectives of the format.

b. To be implementable across a wide range of computer systems without reduction of available features.

c. To provide extensibility to accommodate data types and functional requirements not foreseen.

d. To provide useful capability with limited data formatting overhead.

5. DETAILED REQUIREMENTS

5.1 Format description.

5.1.1 Fixed fields. The format contains header, subheader, and data fields. The NITF header fields are byte aligned. A file header carries information about the identification, classification, structure, content, size of the file as a whole, and size of the major data components within the file. For each kind of data supported by the format, each data item in the file has an associated subheader containing information that describes characteristics of the data item and an associated data field that contains the actual data item.

5.1.2 Extension fields. Flexibility to add support for kinds of data and data characteristics not explicitly defined in this standard is provided within the format. This is accomplished by providing for one or two fields in each header/subheader containing "tagged records" and a group of "tagged data segments." The tagged records in the headers/subheaders may contain additional characteristics of the corresponding data, while the tagged data segments are intended primarily to provide a vehicle for adding support for new kinds of data. The "tags" for the tagged records, and tagged segments, will be coordinated centrally in accordance with MIL-HDBK-1300A to avoid conflicting use, and in some cases, record formats will be configuration managed to control changes to data formats affecting a broad NITF user base.

5.1.3 Supported data types. A NITF file shall support inclusion of four standard kinds of data in a single file: image, symbol, label, and text. It shall be possible to include zero, one, or multiple data items of each standard data type in a single file (for example: several images, but no symbols). Standard data types shall be placed in the file in the following order: all image data items (images), followed by all symbol data items (symbols), followed by all label data items (labels), followed by all text data items (documents). Additional kinds of data may be included in a NITF file by use of Data Extension Segments (DES) (see 5.9). A data item of a standard data type is called a standard data item. A data item of a type defined in a DES is called an extension data item. The order of these major file components is illustrated on figure 2.

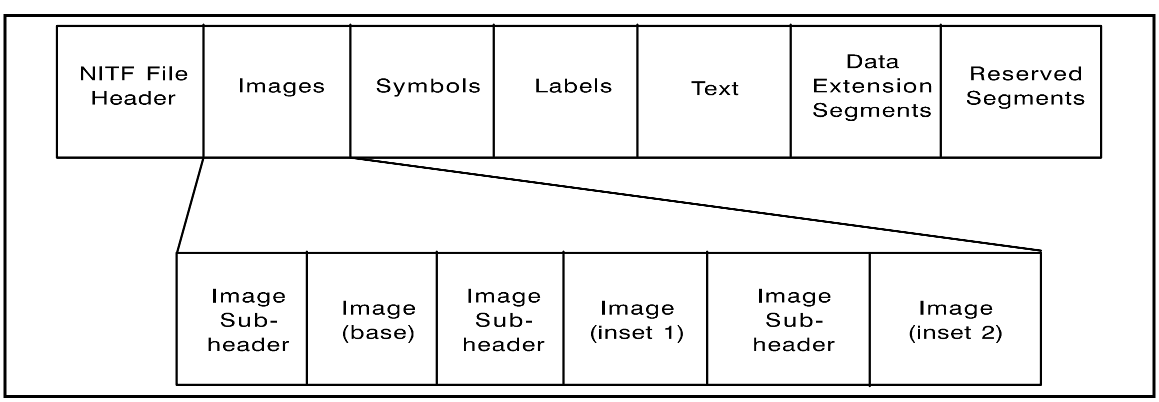


FIGURE 2. NITF file structure.

5.1.4 Application guidance. The NITF file format is intended for typical use with a single image, the base image, and data related to that image. The additional images that may be present in the file are expected to be related images, perhaps enhanced or modified subimages of the base image. These additional images typically will be displayed as "insets", that is, overlaid on the base image. Conceptually, though not required physically by the format, the base image is the "first" image, followed by its overlaid insets. Similarly, labels and symbols are expected to identify or explain information in the base image or one of the subimages. Finally, analysis, background, or other information relevant to interpreting the information content of the base image and its related subimages may be contained in documents - text data items - included in the NITF file. While the format is sufficiently general to support more ambitious applications, it is well to keep the intended use in mind when reading the requirements. NITFS certification requirements (JIEO Circular 9008) for systems will reflect the usage just described.

5.1.5 Standard data item subheaders. Each individual, standard data item included in a NITF file, such as an image or a symbol, shall be preceded by a "subheader" corresponding to that data item. This subheader shall contain information pertaining to that particular data item and data type only. If no items of a given type are included in the file, a subheader for that data type shall not be included in the file. All data items and associated subheaders of a single type shall precede the first subheader for the next data type. The ordering of multiple data items of one type is arbitrary. A diagram of the overall NITF file structure is shown on figure 2 as an example in which there is a "base" image and two smaller "inset" images overlaid on the base. This example is typical of the applications that historically motivated development of the NITF. The expansion of the Images section illustrates the interleaving of multiple images and their associated headers.

5.2 The NITF file header. Each NITF file shall begin with a header, the file header, whose fields contain identification and origination information, file-level security information, and the number and size of data items of each type contained in the file. Figure 3 depicts the NITF file header. It depicts the types of information contained in the header and shows the headers organization as a sequence of groups of related fields. The expansion of the "Image Group" illustrates how the header's overall length and content may expand or contract depending on the number of data items of each type included in the file. The fields of the NITF header are detailed in table I. Definitions of valid data for each header field are detailed in table II.

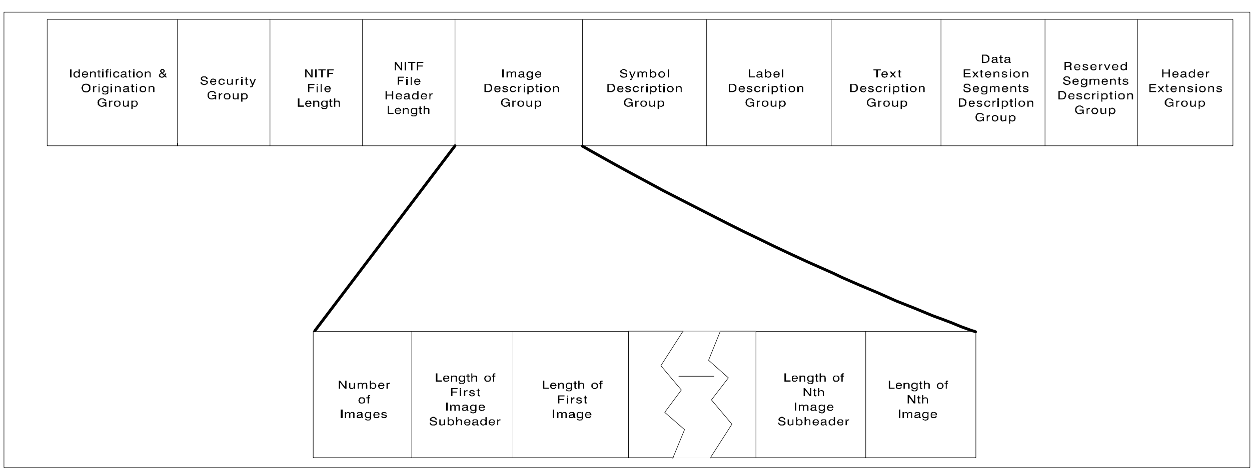


FIGURE 3. NITF file header structure.

TABLE I. NITF file header.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
FHDR	File Type & Version	9	NITFNN.NN	R
CLEVEL	Compliance Level	2	1-99	R
STYPE	System Type	4	Reserved	O
OSTAID	Originating Station ID	10	Alphanumeric	R
FDT	File Date & Time	14	DDHHMMSSZMONYY	R
FTITLE	File Title	80	Alphanumeric	O
FSCLAS	File Security Classification	1	T, S, C, R, or U	R
FSCODE	File Codewords	40	Alphanumeric	O
FSCTLH	File Control and Handling	40	Alphanumeric	O
FSREL	File Releasing Instructions	40	Alphanumeric	O
FSCAUT	File Classification Authority	20	Alphanumeric	O
FSCTLN	File Security Control Number	20	Alphanumeric	O
FSDWNG	File Security Downgrade	6	Alphanumeric	O
FSDEVT	File Downgrading Event	40	Alphanumeric	C
FSCOP	Message Copy Number	5	0-99999	O
FSCPYS	Message Number of Copies	5	0-99999	O
ENCRYP	Encryption	1	0=Not Encrypted 1=Encrypted	R
ONAME	Originator's Name	27	Alphanumeric	O
OPHONE	Originator's Phone Number	18	Alphanumeric	O
FL	File Length	12	0-999999999999	R
HL	NITF File Header Length	6	0-999999	R
NUMI	Number of Images	3	0-999	R
LISH001	Length of 1st Image Subheader	6	0-999999	C
LI001	Length of 1st Image	10	0-9999999999	C
.....				

TABLE I. NITF file header - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
LISHnnn	Length of Nth Image Subheader	6	0-999999	C
LInnn	Length of Nth Image	10	0-9999999999	C
NUMS	Number of Symbols	3	0-999	R
LSSH001	Length of 1st Symbol Subheader	4	0-9999	C
LS001	Length of 1st Symbol	6	0-999999	C
.....				
LSSHnnn	Length of Nth Symbol Subheader	4	0-9999	C
LSnnn	Length of Nth Symbol	6	0-999999	C
NUML	Number of Labels	3	0-999	R
LLSH001	Length of 1st Label Subheader	4	0-9999	C
LL001	Length of 1st Label	3	0-320	C
.....				
LLSHnnn	Length of Nth Label Subheader	4	0-9999	C
LLnnn	Length of Nth Label	3	0-320	C
NUMT	Number of Text Files	3	0-999	R
LTSH001	Length of 1st Text Subheader	4	0-9999	C
LT001	Length of 1st Text File	5	0-99999	C
.....				
LTSHnnn	Length of Nth Text Subheader	4	0-9999	C
LTnnn	Length of Nth Text File	5	0-99999	C
NUMDES	Number of Data Extension Segments	3	0-999	R
LDSH001	Length of 1st Data Extension Segment Subheader	4	0-9999	C
LD001	Length of 1st Data Extension Segment Data Field	9	0-999999999	C
.....				

TABLE I. NITF file header - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
LDSHnnn	Length of n^{th} Data Extension Segment Subheader	4	0-9999	C
LDnnn	Length of n^{th} Data Extension Segment Data Field	9	0-999999999	C
NUMRES	Number of Reserved Extension Segments	3	0-999	R
LRSH001	Length of 1^{st} Reserved Extension Segment Subheader	4	0-9999	C
LR001	Length of 1^{st} Reserved Extension Segment Data Field	7	0-9999999	C
.....				
LRSHnnn	Length of n^{th} Reserved Extension Segment Subheader	4	0-9999	C
LRnnn	Length of n^{th} Reserved Extension Segment Data Field	7	0-9999999	C
UDHDL	User Defined Header Data Length	5	0-99999	R
UDHOFL	User Defined Header Overflow	3	0-999	C
UDHD	User Defined Header Data	*	Registered Tagged Record Extensions	C
XHDL	Extended Header Data Length	5	0-99999	R
XHD	Extended Header Data	**	Controlled Tagged Record Extensions	C

[*]As specified in UDHDL
[**]As specified in XHDL

TABLE II. NITF file header fields.

FHDR	An ASCII character string of the form NITFNN.NN, which indicates this file is formatted using version NN.NN of NITF. The valid values for this field are NITF01.10 and NITF02.00.
CLEVEL	This field shall contain the compliance level required to interpret fully all components of the file. Valid entries are integer values 01 through 06 and 99 and are assigned in accordance with certification requirements established in JIEO Circular 9008. Values 00, and 07 through 98 are reserved for future use.
STYPE	System type or capability. This field is reserved for future use and shall be filled with spaces (ASCII 32, decimal).
OSTAID	This field shall contain the identification code of the originating station.
FDT	This field shall contain the time (Zulu) of the files origination in the format DDHHMMSSZMONYY, where DD is the day of the month (01-31), HH is the hour (00-23), MM is the minute (00-59), SS is the second (00-59), the character Z is required, MON is first three characters of the month; and YY is the last two digits of the year.
FTITLE	This field shall contain the title of the NITF file.
FSCLAS	This field shall contain a valid value representing the classification level of the entire file. Valid values are T (=Top Secret), S (=Secret), C (=Confidential), R (= Restricted), U (=Unclassified).
FSCODE	This field shall contain a valid indicator of the security compartments associated with the file. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, and complete codewords or project numbers. The selection of a relevant set of codewords and project numbers is application specific. If this field is all spaces, it shall imply that no codewords apply to the file.
FSCTLH	This field shall contain valid security handling instructions associated with the file. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, complete codewords or project numbers, complete words and abbreviations of more than two characters, phrases only if the words within the phrase are separated by hyphens. The selection of a relevant set of security handling instructions is implementation specific. If this field is all spaces, it shall imply that no file control and handling instructions apply.

TABLE II. NITF file header fields - Continued.

FSREL	This field shall contain a valid list of countries and/or groups of countries to which the file is authorized for release. Valid items in the list are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: country codes and groupings that are digraphs in accordance with FIPS PUB 10-3. If this field is all spaces, it shall imply that no file release instructions apply.
FSCAUT	This field shall contain a valid identity code of the classification authority for the file. The code shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no file classification authority applies.
FSCTLN	This field shall contain a valid security control number associated with the file. The format of the security control number shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no file security control number applies.
FSDWNG	This field shall contain a valid indicator that designates the point in time at which a declassification or downgrading action is to take place. The valid values are (1) the calendar date in the format YYMMDD, (2) the code "999999" when the originating agency's determination is required (OADR), and (3) the code "999998" when a specific event determines at what point declassification or downgrading is to take place. If this field is all spaces, it shall imply that no file security downgrade condition applies.
FSDEVT	If the File Security Downgrade field (FSDWNG) equals "999998," this field shall be present and shall contain a valid specification of the downgrade event. If this field is present and all spaces, it shall imply that an error exists. Valid values for the event specification are determined by the application.
FSCOP	This field shall contain the copy number of the file.
FSCPYS	This field shall contain the total number of copies of the file.
ENCRYP	This field shall contain the value zero until such time as this specification is updated to define the use of other values.
ONAME	This field shall contain a valid name for the operator who originated the file. If the field is all spaces, it shall mean that no operator is assigned responsibility for origination.
OPHONE	If not all blanks, this field shall contain a valid phone number for the operator who originated the file. If the field is all spaces, it shall mean that no phone number is available for the operator assigned responsibility for origination.
FL	This field shall contain the length in bytes of the entire file including all headers. subheaders, and data. The value of this field never shall be zero.
HL	This field shall contain a valid length in bytes of the NITF file header. The value of this field never shall be zero.

TABLE II. NITF file header fields - Continued.

NUMI	This field shall contain the number of separate images included in the file. The value is valid only if it is within the specified range and the total of the Number of Images, plus the Number of Symbols, plus the Number of Labels does not exceed 999. This field shall be zero if and only if no images are included in the file.
LISH001	If the field NUMI contains a value of one or more, this field shall contain a valid length in bytes for the subheader of the first image in the file. This field is conditional and shall be omitted if NUMI field contains zero.
LI001	This field shall contain a valid length in bytes of the first image. This field is conditional and shall be omitted if NUMI field contains zero.
LISHnnn	This field shall contain a valid length in bytes for the nnnth image subheader, where nnn is the number of the image counting from the first image (nnn=001) in order of the images' appearance in the file. This field shall occur as many times as specified in the NUMI field. This field is conditional and shall be omitted if NUMI field contains zero.
LInnn	This field shall contain a valid length in bytes of the nnnth image, where nnn is the image number of the image counting from the first image (nnn=001) in order of the images' appearance in the file. If the image is compressed, the length after compression shall be used. This field shall occur as many times as specified in the NUMI field. This field is conditional and shall be omitted if NUMI field contains zero.
NUMS	This field shall contain the number of separate symbols included in the file. The value is valid only if it is within the specified range and the total of the Number of Images, plus the Number of Symbols, plus the Number of Labels does not exceed 999. This field shall be zero if and only if no symbols are included in the file.
LSSH001	If the field NUMS contains a value of one or more, this field shall contain a valid length in bytes for the subheader of the first symbol in the file. This field is conditional and shall be omitted if NUMS field contains zero.
LS001	This field shall contain a valid length in bytes for the first symbol. This field is conditional and shall be omitted if NUMS field contains zero.
LSSHnnn	This field shall contain a valid length in bytes for the nnnth symbol subheader, where nnn is the number of the symbols counting from the first symbol (nnn=001) in the order of the symbols' appearance in the file. This field shall occur as many times as specified in the NUMS field. This field is conditional and shall be omitted if NUMS contains zero.
LSnnn	This field shall contain a valid length in bytes of the nnnth symbol, where nnn is the symbol number of the symbol, counting from the first symbol (nnn=001) in the order of the symbols' appearance in the file. This field shall occur as many times as specified in the NUMS field. This field is conditional and shall be omitted if NUMS field contains zero.

TABLE II. NITF file header fields - Continued.

NUML	This field shall contain the number of separate labels included in the file. The value is valid only if it is within the specified range, and the total of the Number of Images, plus the Number of Symbols, plus the Number of Labels does not exceed 999. This field shall be zero if and only if no labels are included in the file.
LLSH001	If the field NUML contains a value of one or more, this field shall contain a valid length in bytes for the subheader of the first label in the file. This field is conditional and shall be omitted if NUML field contains zero.
LL001	This field shall contain a valid length in bytes for the first label. This field is conditional and shall be omitted if NUML field contains zero.
LLSHnnn	This field shall contain a valid length in bytes for the nnnth label subheader, where nnn is the number of the labels, counting from the first label (nnn=001) in the order of the labels' appearance in the file. This field shall occur as many times as specified in the NUML field. This field is conditional and shall be omitted if NUML field contains zero.
LLnnn	This field shall contain a valid length in bytes of the nnnth label, where nnn is the label number of the label, counting from the first label (nnn=001) in order of the labels' appearance in the file. This field shall occur as many times as specified in the NUML field. This field is conditional and shall be omitted if NUML field contains zero.
NUMT	This field shall contain the number of separate text items included in the file. The value is valid only if it is within the specified range. This field shall be zero if and only if no text items are included in the file.
LTSH001	If the field NUMT contains a value of one or more, this field shall contain a valid length in bytes for the subheader of the first text item in the file. This field is conditional and shall be omitted if NUMT field contains zero.
LT001	This field shall contain a valid length in bytes for the first text item. This field is conditional and shall be omitted if NUMT field contains zero.
LTSHnnn	This field shall contain a valid length in bytes for the nnnth text item subheader, where nnn is the number of the text item, counting from the first text item (nnn=001) in the order of the text items' appearance in the file. This field shall occur as many times as specified in the NUMT field. This field is conditional and shall be omitted if NUMT field contains zero.
LTnnn	This field shall contain a valid length in bytes of the nnnth text item, where nnn is the number of the text item, counting from the first text item (nnn=001) in the order of the text items' appearance in the file. This field shall occur as many times as specified in the NUMT field. This field is conditional and shall be omitted if NUMT field contains zero.

TABLE II. NITF file header fields - Continued.

NUMDES	This field shall contain the number of separate data extension segments included in the file. This field shall be zero only if no data extension segments are included in the file.
LDSH001	If the field NUMDES contains a value of one or more, this field shall contain a valid length in bytes for the subheader of the first data extension segment in the file. This field is conditional and shall be omitted if the NUMDES field contains zero.
LD001	This field shall contain a valid length in bytes for the data field of the first data extension segment. This field is conditional and shall be omitted if the NUMDES field contains zero.
.....	
LDSHnnn	This field shall contain a valid length in bytes for the nnnth extension segment subheader, where nnn is the number of the data extension segment counting from the first data extension segment (nnn=001) in order of the data extension segments appearance in the file. This field shall occur as many times as are specified in the NUMDES field. This field is conditional and shall be omitted if the NUMDES field contains zero.
LDnnn	This field shall contain a valid length in bytes of the data field of the nnnth data extension segment, where nnn is the image number of the data extension segment counting from the first data extension segment (nnn=001) in order of the data extension segments appearance in the file. This field shall occur as many times as are specified in the NUMDES field. This field is conditional and shall be omitted if the NUMDES field contains zero.
NUMRES	This field shall contain the number of separate reserved extension segments included in the file. This field shall be zero until such time as one or more reserved extension segments is defined.
LRESH001	This field is conditional and shall be omitted if the NUMRES field contains zero.
LRE001	This field is conditional and shall be omitted if the NUMRES field contains zero.
.....	
LRESHnnn	This field is conditional and shall be omitted if the NUMRES field contains zero.
LREnnn	This field is conditional and shall be omitted if the NUMRES field contains zero.
UDHDL	This field shall contain the length in bytes of the entire UDHD field. The length is three plus sum of the lengths of all the registered tagged record extensions (see 5.9.1.1) appearing in the UDHD field, since they are not separated from one another. A value of zero shall mean that no registered tagged record extensions are included in the header. If a registered tagged record extension is too long to fit in the UDHD field, it may be put in a data extension segment (see 5.9.1.3.1).

TABLE II. NITF file header fields - Continued.

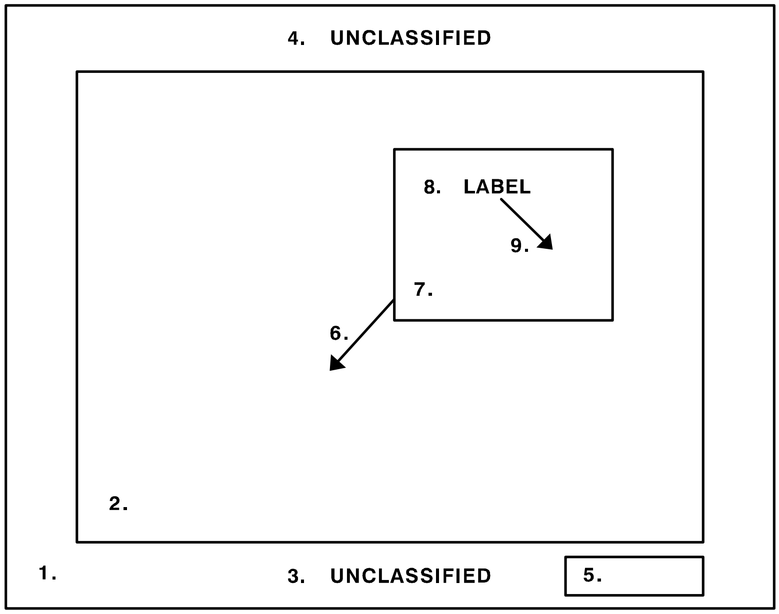
UDHOFL	If present, this field shall contain "000" if the tagged record extensions in UDMD do not overflow into a DES, or shall contain the sequence number of the DES into which they do overflow, this field shall be omitted if the field UDHDL contains zero.
UDHD	If present, this field shall contain user defined registered tagged record extensions (see 5.9.1.1). The length of this field shall be the length specified by the field UDHDL. Registered tagged record extensions shall appear one after the other with no intervening bytes. The first byte of this field shall be the first byte of the first registered tagged record extension appearing in the field. The last byte of this field shall be the last byte of the last registered tagged record extension to appear in the field. This field shall be omitted if the field UDHDL contains zero.
XHDL	This field shall contain the length in bytes of the entire XHD field. The length is three plus sum of the lengths of all the controlled tagged record extensions (see 5.9.1.2) appearing in the XHD field, since they are not separated from one another. A value of zero shall mean that no controlled tagged record extensions are included in the NITF header. If a controlled tagged record extension is too long to fit in the XHD field, it may be put in a data extension segment (see 5.9.1.3.1).
XHD	If present, this field shall contain controlled tagged record extensions (see 5.9.1.2) approved and under configuration management of the NITF Technical Board. The length of this field shall be the length specified by the field XHDL. The first three characters shall be "000" if this field does not overflow into a DES, or shall contain the sequence number in the file of the DES into which it does overflow. Controlled tagged record extensions shall appear one after the other with no intervening bytes. The fourth byte of this field shall be the first byte of the first controlled tagged record extension appearing in the field. The last byte of this field shall be the last controlled tagged record extension to appear in the field. This field shall be omitted if the field XHDL contains zero.

5.3 NITF overlay concept. The NITF is sufficiently flexible to support future enhancements while remaining backward compatible. However, this flexibility also allows applications to use the NITF in ways that do not reflect the underlying need that the NITF was developed to address. Such applications, while possibly conforming to the letter of the NITF file format specification, may not be certifiable or accreditable for use by U.S. Government organizations. (For detailed NITFS certification requirements, the reader should consult the JIEO Circular 9008.) The following subsections describe relationships anticipated to exist among the data items in an NITF file and how these relationships are represented in the file.

5.3.1 Image product/file relationships. Though the concept of an image product may include multiple files in the future, it is expected that typical applications will represent an image product in a single file. Within each image product the image with the lowest display level is the base image. Each image product shall comprise one base image plus associated data. If a base image is present, it shall form the basis for using the other data contained in the product. Images other than the base image are inset images. Inset images contained in the product are intended to be referenced to the base image, possibly by their placement (via the ILOC field of the image subheader) relative to the base image or by visual cues provided by symbols and labels. All other images (the "inset" images), symbols, and labels are expected to define overlays to the base image in the sense that, when displayed, they will overwrite the base image. The relative visibility, when displayed, of the various displayable items in the file is recorded in the file by use of the display level (the "DLVL" field in the standard data type subheaders, specifically IDLVL for images, SDLVL for symbols, and LDLVL for labels). Groupings of related items may be formed by use of the attachment level (the "ALVL" field in the standard data type subheaders, specifically IALVL for images, SALVL for symbols, and LALVL for labels). The aggregate of the data items in an NITF file, including extended data as described in 5.9, should be regarded as constituting a single image-based product. Although loose aggregations of items of the various supported data types having no particular relationship to one another could be put into an NITF file, this use of the format would conflict with the motivations behind the NITF development. Use of the format in such a way is strongly discouraged.

5.3.2 Overlays and display level. The order in which images, symbols, and labels are "stacked" visually when displayed shall be determined by their display level (the DLVL field in the standard data type subheaders, specifically IDLVL for images, SDLVL for symbols, and LDLVL for labels), not by their relative position within the NITF file. The display level is a positive integer less than 1000. Every image, symbol, and label in an NITF file shall have a unique display level. That is, no two items may have the same display level. This requirement allows display appearance to be independent of data processing order.

5.3.3 Display level interpretation. The display level determines the display precedence of images, symbols, and labels when they are output to a display device. That is, at any pixel location shared by more than one image, symbol, or label, the value displayed there is that determined from the item with the highest numbered display level. An example is provided on figure 4. Figure 4 illustrates a sample "output presentation" from an NITF file that illustrates the effects of display level assignment. The Display Level (DL) of each item shown on figure 4 is indicated in the list of items on figure 4, where the list is in the order that the items were placed in the NITF file containing them. In the case shown, the item with display level one is not an image but rather an opaque CGM rectangle (symbol data, not image data). Because the CGM rectangle is larger than the base image (which, in this case, serves as the first overlay because its display level is two), it provides a border to the base image. Following increasing DL value, the border is overlaid by the exploited image which, in turn, is overlaid by arrow one, which is in turn overlaid by the image inset, which is overlaid by the label, which is overlaid by the arrow label, etc. It is emphasized again that data are not displayed in the same sequence in which they appear in the NITF file. The AL values in the list refer to "Attachment Levels" these are described next.



1. Border (an opaque box)	DL = 001	AL = 000
2. Exploited Image	DL = 002	AL = 001
3. Classification Marking 1	DL = 999	AL = 000
4. Classificaton Marking 2	DL = 998	AL = 000
5. Handling Instructions	DL = 997	AL = 000
6. Arrow 1 (image inset)	DL = 003	AL = 002
7. Image Inset	DL = 004	AL = 002
8. Label	DL = 005	AL = 004
9. Arrow 2 (label)	DL = 006	AL = 005

FIGURE 4. Display level illustration.

5.3.4 Attachment level. Attachment level (AL) provides a way to associate display items (images, symbols, and labels) with one another so they may be treated together for certain operations such as moving them, rotating them, or displaying them as a group. The attachment level of a display item shall be equal to the display level of the display item to which it is "attached." This value is stored in the "ALVL" field (specifically IALVL for images, SALVL for symbols, and LALVL for labels) of the item's subheader. An item with Display Level 1 (the minimum display level in this example), must have an attachment level of zero. An attachment level of zero shall be interpreted as "unattached." The item having minimum display level shall have attachment level zero and location (0,0). Any other item may also have AL zero, that is, be unattached. An overlay's display level shall always be numerically greater than its attachment level (that is, an overlay must be attached to something previously displayed or is unattached). Figure 5 shows the attachment relationships of the overlays on figure 4. When an overlay or base is edited (moved deleted, rotated), all overlays attached to it, directly or indirectly, may be affected by the same operation. For example, on figure 5, if the exploited image (DL 002, AL 001) were moved one centimeter to the left, the arrows (DL 003, AL 002, and DL 006, AL 005), label (DL 005, AL 004), and the image inset (DL 004, AL 002) associated with the exploited image (DL 002, AL 001) also would be moved one centimeter to the left. Also note that because of the way the attachments have been constructed, if the label (DL 005, AL 004) were deleted, so would be its associated arrow 2 (006, 005). However, if the image inset (DL 004, AL 002) were deleted, its associated arrow 1 (DL 003, AL 002) would not be deleted.

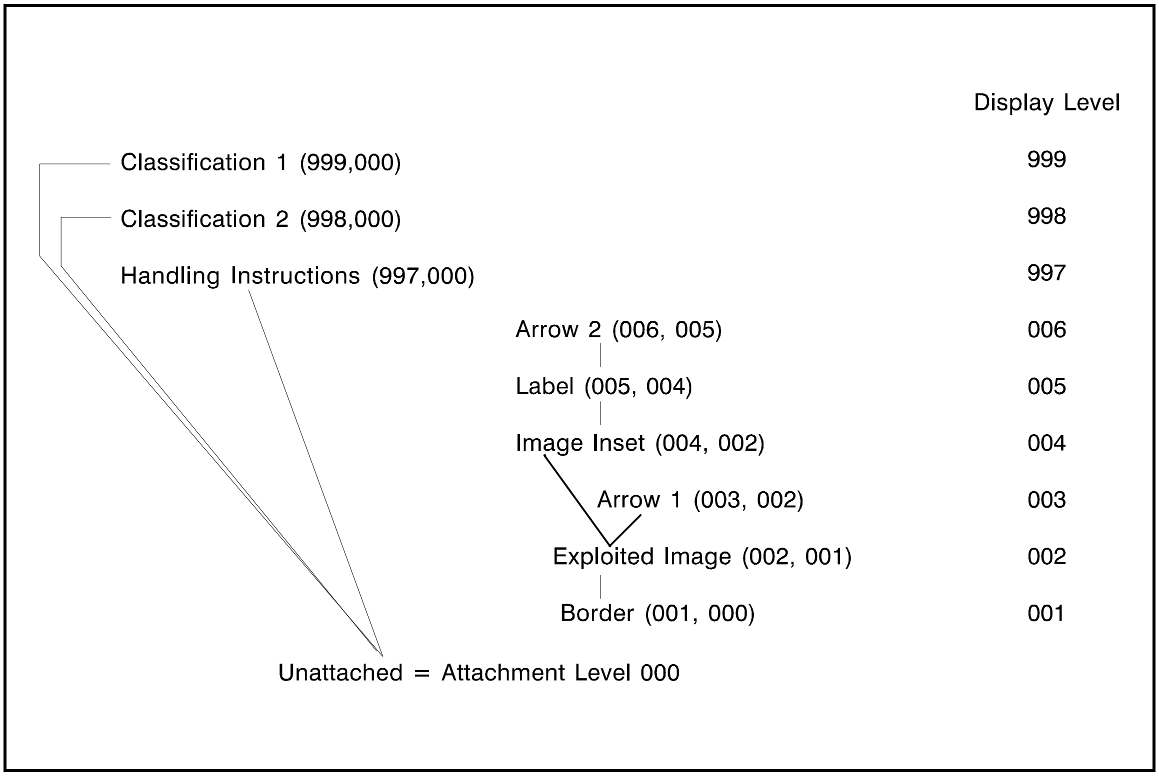


FIGURE 5. Attachment level relationships.

5.4 Header field specification. The specification of the fields in the various headers potentially found within an NITF file is provided in a series of tables. Each header uses two tables for its description. For each field in the header, the first table includes a mnemonic identifier, the field's name, the field size, the range of values it may contain, and an indication of its "type" (see 5.4.2). The second table contains a description of the valid contents of the field and any constraints on the field's use. The NITF file header fields are specified in tables I and II. The standard data item subheader fields are specified in tables III, IV, and IX through XIV. The tagged record extension headers (see 5.9.1 and 5.9.1.1) are defined in tables XV and XVI. Finally, the data extension segment header fields (see 5.9.1.2) are defined in tables XVII and XVIII. The data that appears in all header information fields specified in the tables, including numbers, shall be represented using the printable ASCII character set (defined in Appendix A) with eight bits (one byte) per character. Representing numbers in character form avoids many of the problems associated with differences in word length and internal representation among different machines. Representing the header and subheader fields in ASCII also makes them more easily read by humans. All field size specifications given for the header and subheader fields specify a number of bytes. Fields that may contain any printable ASCII characters (including punctuation marks) are indicated as "Alphanumeric" in the

Value Range specification. The reader is warned that this is a nonstandard use of the term alphanumeric. The allowable range of values for numeric fields typically is indicated in the form N-M, where N and M are the minimum and maximum values, respectively.

TABLE III. NITF image subheader.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
IM	File Part Type	2	IM	R
IID	Image ID	10	Alphanumeric	R
IDATIM	Image Date & Time	14	DDHHMMSSZMONYY	O
TGTID	Target ID	17	BBBBBBBBBBFFFFFCC	O
ITITLE	Image Title	80	Alphanumeric	O
ISCLAS	Image Security Classification	1	T, S, C, R, or U	R
ISCODE	Image Codewords	40	Alphanumeric	O
ISCTLH	Image Control and Handling	40	Alphanumeric	O
ISREL	Image Releasing Instructions	40	Alphanumeric	O
ISCAUT	Image Classification Authority	20	Alphanumeric	O
ISCTLN	Image Security Control Number	20	Alphanumeric	O
ISDWNG	Image Security Downgrade	6	Alphanumeric	O
ISDEVT	Image Downgrading Event	40	Alphanumeric	C
ENCRYP	Encryption	1	0=Not Encrypted 1=Encrypted	R
ISORCE	Image Source	42	Alphanumeric	O
NROWS	Number of Significant Rows in image	8	0-99999999	R
NCOLS	Number of Significant Columns in image	8	0-99999999	R
PVTYPE	Pixel value type	3	Alphanumeric	R

TABLE III. NITF image subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
IREP	Image Representation	8	Alphanumeric	R
ICAT	Image Category	8	Alphanumeric	R
ABPP	Actual Bits-Per-Pixel Per Band	2	01-64	O
PJUST	Pixel Justification	1	L or R	O
ICORDS	Image Coordinate System	1	U, G, C, or N	R
IGEOLO	Image Geographic Location	60	ddmmssXdddmmssY (four times) or, ggXYZmmmmmmmmmm (four times)	C
NICOM	Number of Image Comments	1	0-9	R
ICOM1	Image Comment 1	80	Alphanumeric	C
.....				
ICOMnn	Image Comment nn	80	Alphanumeric	C
IC	Image Compression	2	NC, NM, C0, C1, C2, C3, C4, M0, M1, M2, M3, or M4	R
COMRAT	Compression Rate Code	4	Alphanumeric	C
NBANDS	Number of Bands	1	1-9	R
IREPBAND1	1st Band Representation	2	Alphanumeric	R
ISUBCAT1	1st Band Significance for Image Category	6	Alphanumeric	R
IFC1	1st Band Image Filter Condition	1	N	R
IMFLT1	1st Band Standard Image Filter Code	3	Reserved	R
NLUTS1	1st Band Number of LUTS	1	0-4	R

TABLE III. NITF image subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
NELUT1	1st Band Number of LUT Entries	5	1-65536	C
LUTD1	Data of 1st LUT for 1st Band	†	LUT Values	C
.....				
LUTDmm	Data of mmth LUT for 1st Band	†	LUT Values	C
.....				
IREPBANDnn	nnth Band Representation	2	Alphanumeric	C
ISUBCATnn	nnth Band Sub Category	6	Alphanumeric	C
IFCnn	nnth Band Image Filter Condition	1	N	C
IMFLTnn	nnth Band Standard Image Filter Code	3	Reserved	C
NLUTSnn	nnth Band Number of LUTS	1	0-4	C
NELUTnn	nnth Band Number of LUT Entries in	5	1-65536	C
LUTD1	nnth Band Data of 1st LUT	†	LUT Values	C
.....				
IREPBANDnn	nnth Band Representation	2	Alphanumeric	C
ISUBCATnn	nnth Band Sub Category	6	Alphanumeric	C
IFCnn	nnth Band Image Filter Condition	1	N	C
IMFLTnn	nnth Band Standard Image Filter Code	3	Reserved	C
NLUTSnn	nnth Band Number of LUTS	1	0-4	C
NELUTnn	nnth Band Number of LUT Entries in	5	1-65536	C

TABLE III. NITF image subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
LUTD1	nnth Band Data of 1st LUT	†	LUT Values	C
.....				
LUTDmm	nnth Band Data of the mmth LUT	†	LUT Values	C
ISYNC	Image Sync Code	1	0 or 4	R
IMODE	Image Mode	1	B, P, S	R
NBPR	Number of Blocks Per Row	4	1-9999	R
NBPC	Number of Blocks Per Column	4	1-9999	R
NPPBH	Number of Pixels Per Block Horizontal	4	1-9999	R
NPPBV	Number of Pixels Per Block Vertical	4	1-9999	R
NBPP	Number of Bits Per Pixel Per Band	2	1-64	R
IDLVL	Display Level	3	1-999	R
IALVL	Attachment Level	3	0-998	R
ILOC	Image Location	10	rrrrrccccc	R
IMAG	Image Magnification	4	Alphanumeric	R
UDIDL	User Defined Image Data Length	5	0-99999	R
UDOFL	User Defined Overflow	3	0-999	C
UDID	User Defined Image Data	*	Registered Tagged Record Extensions	C
IXSHDL	Extended Subheader Data Length	5	0-99999	R

TABLE III. NITF image subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
IXSOFL	Extended Subheader Overflow	3	0-999	C
IXSHD	Extended Subheader Data	**	Controlled Tagged Record Extensions	C

† One Byte for each entry
* As specified in UDIDL
** As specified in IXSHDL

TABLE IV. NITF image subheader fields.

IM	This field shall contain the characters "IM" to identify the subheader as an image subheader.
IID	This field shall contain a valid alphanumeric identification code associated with the image. The valid codes are determined by the application.
IDATIM	This field shall contain the time (Zulu) of acquisition of the image in the format DDHHMMSSZMONYY, in which DD is the day of the month (01-31); HH is the hour (00-23); MM is the minute (00-59); SS is the second (00-59); the character Z is required; MON is the first three characters of the month; and YY is the last two digits of the year.
TGTID	This field shall contain the identification of the primary target in the format, BBBBBBBBBBFFFFFCC, consisting of ten characters of BE (Basic Encyclopedia) identifier, followed by five characters of functional category code, followed by the two character country code as specified in FIPS PUB 10-3.
ITITLE	This field shall contain the title of the image.
ISCLAS	This field shall contain a valid value representing the classification level of the image. Valid values are: T (=Top Secret), S (=Secret), C(=Confidential), R (=Restricted), U (=Unclassified).
ISCODE	This field shall contain a valid indicator of the security compartments associated with the image. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, and complete codewords or project numbers. The selection of a relevant set of codewords and project numbers is specific to the application. If this field is all spaces, it shall imply that no codewords apply to the image.

TABLE IV. NITF image subheader fields - Continued.

ISCTLH	This field shall contain valid security handling instructions associated with the image. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, complete codewords and abbreviations of more than two characters, and phrases only if the words within the phrase are separated by hyphens. The selection of a relevant set of security handling instructions is implementation specific. If this field is all spaces, it shall imply that no image control and handling instructions apply.
ISREL	This field shall contain a valid list of countries and/or groups of countries to which the image is authorized for release. Valid items in the list are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: country codes and groupings that are digraphs in accordance with FIPS PUB 10-3. If this field is all spaces, it shall imply that no image release instructions apply.
ISCAUT	This field shall contain a valid identity code of the classification authority for the image. The code shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no image classification authority applies.
ISCTLN	This field shall contain a valid security control number associated with the image. The format of the security control number shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no image security control number applies.
ISDWNG	This field shall contain a valid indicator that designates the time at which a declassification or downgrading action is to take place. The valid values are (1) the calendar date in the format YYMMDD, (2) the code "999999" when the originating agency's determination is required (OADR), and (3) the code "999998" when a specific event determines at what point in time declassification or downgrading is to take place. If this field is all spaces, it shall imply that no image security downgrade condition applies.
ISDEVT	If the Image Security Downgrade field (ISDWNG) equals "999998," this field shall be present and shall contain a valid specification of the downgrade event. If this field is present and contains all spaces, it shall imply that an error exists. Valid values for the event specification are determined by the application.
ENCRYP	This field shall contain the value zero until such time as this specification is updated to define the use of other values.
ISORCE	If not blanks, this field shall contain a description of the source of the image. If the source of the data is classified, then the description shall be preceded by the classification, including codeword(s) (table V). Valid data are alphanumeric text.

TABLE IV. NITF image subheader fields - Continued.

NROWS	This field shall contain the total number of rows of significant pixels in the image. When NPPBV * NBPC > NROWS, the remaining last rows (NPPBV * NBPC - NROWS) shall contain fill data (such as, only the rows indexed 0 through NROWS - 1 of the image contain "significant" data). The pixel fill values are determined by the application.
NCOLS	This field shall contain the total number of columns of significant pixels in the image. When NPPBH * NBPR > NCOLS, the remaining last pixels of each row (NPPBH * NBPR - NCOLS) shall contain fill data (that is, only the columns indexed 0 through NCOLS - 1 of the image contain "significant data). The pixel fill values are determined by the application.
PVTYPE	This field shall contain an indicator of the type of computer representation used for the value for each pixel for each band in the image. Valid entries are INT for integer, SI for 2's complement signed integer, R for real, C for complex, B for bit-mapped, U for user-defined. The data bits of INT and SI values shall appear in the file in order of significance, beginning with the most significant bit (MSB) and ending with the least significant bit (LSB). INT and SI data types shall be limited to 16 bits. R values shall be represented according to IEEE 32-bit floating point representation. C values shall be represented with the Real and Imaginary parts each represented in IEEE 32-bit floating point representation and appearing in adjacent four-byte blocks, first Real, then Imaginary. B pixel values shall be represented as single bits with value 1 or 0.
IREP	This field shall contain a valid indicator for the general kind of image represented by the data. Valid representation indicators are MONO for monochrome; RGB for red, green, or blue true color, RGB/LUT for mapped color; and MULTI for multiband imagery. In addition, compressed imagery can have this field set to YCbCr601 when compressed in the CCIR 601 color space using JPEG (field IC=C3). This field should be used in conjunction with the ICAT, ISUBCATnn, and IREPBANDnn fields to interpret the significance of each band in the image.
ICAT	This field shall contain a valid indicator of the specific category (often revealing the nature of the collector or intended use) of imagery. Valid categories are VIS for visible imagery, MAP for maps, SAR for synthetic aperture radar, IR for infrared, MS for multispectral, FP for fingerprints, MRI for magnetic resonance imagery, XRAY for x-rays, and CAT for cat scans. The default value is VIS. This field should be used in conjunction with the IREP, ISUBCATnn, and IREPBANDnn fields to interpret the significance of each band in the image.

TABLE IV. NITF image subheader fields - Continued.

ABPP	This field shall contain the number of "significant bits" for the value in each band of each pixel without compression. Even when the image is compressed, ABPP contains the number of significant bits per pixel that were present in the image before compression. This field shall be less than or equal to Number of Bits Per Pixel (field NBPP). The number of adjacent bits within each NBPP is used to represent the value. These "representation bits" shall be left justified or right justified within the NBPP bits, according to the value in the PJUST field. For example, if 11-bit pixels are stored in 16 bits, this field shall contain 11 and NBPP shall contain 16. The default number of "significant bits" to be used (if this field is all zeros) is the value contained in NBPP.
PJUST	When ABPP is not equal to NBPP, this field indicates whether the significant bits are left justified (L) or right justified (R). Nonsignificant bits in each pixel shall contain the value 0. Any value other than L or R in this field shall indicate right justified.
ICORDS	This field shall contain a valid code indicating the geo-referenced coordinate system for the image. The valid values for this field are: U=UTM, G=Geodetic (Geographic), C=Geocentric, N=None.
IGEOLO	If the Image Coordinate System field ICORDS value is not N, this field shall contain a valid geographic location, in terms of corner locations, of the image in the coordinate system specified in the ICORDS field. The locations of the four corners of the (significant) image data shall be given in image coordinate order: (0,0), (0, MaxCol), (MaxRow, MaxCol), (MaxRow, 0). MaxCol and MaxRow shall be determined from the values contained, respectively, in NCOLS and NROWS as MaxCol = NCOLS - 1 and MaxRow = NROWS - 1. Valid corner locations in geodetic and geocentric coordinates shall be expressed as latitude and longitude. The format ddmmssX represents degrees, minutes, and seconds of latitude with X = N or S for north or south, and dddmmssY represents degrees, minutes, and seconds of longitude with Y = E or W for east or west, respectively. For the UTM coordinate system, coordinates shall be expressed in UTM grid coordinates (also known as Military Grid Reference System (MGRS) coordinates) to the accuracy indicated by the Value Range specification. A description of UTM Grid Coordinates can be found in Technical Manual No. 5-241-1 of the Department of the Army, GRIDS and GRID REFERENCES, 1983.
NICOM	This field shall contain the valid number of free text image comments.
ICOM1	This field shall contain the first line of comment text. The fields ICOM1 through ICOMnn, if present, shall contain free form alphanumeric text. They are intended for use as a single comment block and should be used that way. If the comment is classified, it will be preceded by the classification, including codeword(s). This field shall be omitted if the value in the Number of Image Comments field (NICOM) is zero.

TABLE IV. NITF image subheader fields - Continued.

.....	
ICOMnn	This field shall contain the nnth line of comment text, for 1 < nn $\leq$ value in the NICOM field. See description of ICOM1 for usage. This field shall be omitted if the value in the NICOM field is zero.
IC	This field shall contain a valid code indicating the form of compression used in representing the image data. Valid values for this field are C0, to mean compressed with a user specified algorithm, C1 to mean bi-level, C2 to mean ARIDPCM, C3 to mean JPEG, C4 to mean Vector Quantization and NC to mean the image is not compressed. Also valid are the codes M0, M3 and M4 for compressed images, and NM for uncompressed images, indicating a blocked image that contains a block mask and/or a transparent pixel mask. The format of a mask image is identical to the format of its corresponding non-masked image, except for the presence of an Image Data Mask Subheader at the beginning of the image data area. The format of the Image Data Mask Subheader is described in 5.5.1.5 and is shown in Table IV(A). The definitions of the compression schemes associated with codes C1, C2, C3, and C4 are given, respectively, in MIL-STD-188-196, MIL-STD-188-197A, MIL-STD-188-198A, and MIL-STD-188-199. This field shall not contain C1 or C2 if NBANDS > 1 or NBLOCKS > 1.

TABLE IV. NITF image subheader fields - Continued.

COMRAT	If the Image Compression (IC) field contains C0, C1, C2, C3, C4, M0, M3, or M4, this field shall be present and contain a code indicating the compression rate for the image. If the value in IC is C0 or M0, the code shall be user defined but shall not be all blanks. If the value in IC is C1 or M1, the valid codes are 1D, 2DS, and 2DH, where: 1D means one Dimensional Coding, 2DS means two Dimensional Coding Standard Vertical Resolution, K=2 2DH means two Dimensional Coding High Vertical Resolution, K=4 Explanation of these codes can be found in MIL-STD-188-196. If the value in IC is C2 or M2, this field shall contain a value given in the form n.nn representing the number of bits-per-pixel for the compressed image. Explanation of the compression rate for vector quantization can be found in MIL-STD-188-199. Valid codes in this case are 0.75, 1.40, 2.30, and 4.50. Explanation of these codes can be found in MIL-STD-188-197A. If the value in IC is C3 or M3, this field is used to identify the default quantization table(s) used by the JPEG compression algorithm. In this case, the format of this field is XX.Y where XX is the image data type (00 = general purpose, 01 through 99 are reserved), and Y represents the quality level 1 through 5. Explanation of these codes can be found in MIL-STD-188-198A. If the value in IC is C4 or M4, this field shall contain a value given in the form n.nn representing the number of bits-per-pixel for the compressed image. Explanation of the compression rate for vector quantization can be found in MIL-STD-188-199. This field is omitted if the value in IC is NC or NM.
NBANDS	This field shall contain the number of bands comprising the image. This field and the IREP field are interrelated and independent of the IMODE field. The corresponding values for (IREP, NBANDS) are (MONO, 1); (RGB, 3); (RGB/LUT, 1); (YCbCr601, 3); (MULTI, 2-9).
IREPBAND1	When NBANDS contains the value one, this field shall contain all spaces. In all other cases, this field shall contain a valid indicator of the interpretation of the first band. Valid values are R, G, and B when IREP contains RGB; the band number is a positive integer when IREP contains MULTI. In all other cases, the use of this field is user-defined. However, its purpose is to provide the significance of the first band of the image with regard to the general image type as recorded in IREP. The significance of each band in the image can be derived from the combination of the IREP, IREPBANDnn and ICAT and ISUBCATnn fields.
ISUBCAT1	The use of this field is user-defined. Its purpose is to provide the significance of the first band of the image with regard to the specific category, ICAT, of the overall image. An example would be the wavelength of IR imagery.

TABLE IV. NITF image subheader fields - Continued.

IFC1	This field shall contain the value N (to mean none). Other values are reserved for future use.
IMFLT1	This field is reserved for future use. It shall be filled with blanks.
NLUTS1	This field shall contain the number of look-up tables associated with the 1st band of the image. If the image is a single band (NBANDS = 1), pseudocolor (IREP = RGB/LUT) image, this field shall contain the value three. The first, second, and third LUTS, in this case, shall map the image to the Red, Green, and Blue display bands, respectively. This is not an option for any band after the first (NLUTSnn, nn>1), since RGB/LUT images are single band. Any other use of look-up tables is defined by application.
NELUT1	This file shall contain the number of entries in each of the look-up tables for the first band of data. This field shall be omitted if the value in NLUTS1 is zero.
LUTD1	This field shall be omitted if the first Band Number of LUTs is zero. Otherwise, this field shall contain the data defining the first look-up table for the first image band. Each entry in the look-up table is composed of one byte, ordered from most significant bit to least significant bit representing a value from 0 to 255. To use the look-up table for each integer k, $0 \leq k \leq$NELUT1-1, the pixel value k in the first image band shall be mapped to the value of the k^{th} byte of the look-up table. This field supports only integer band data (PVTYPE = INT).
.....	
LUTD1nn	This field shall be omitted if the 1st Band Number of LUTs is zero. Otherwise, this field shall contain the data defining the nn^{th} look-up table for the first image band. This field shall occur for each nn with $1 < nn \leq$NLUTS1. Each entry in the look-up table is composed of one byte, ordered from most significant bit to least significant bit representing a value from 0 to 255. To use the look-up table, for each integer n,$0 \leq k \leq$NELUT1, the pixel value k in the nn^{th} image band shall be mapped to the value of the k^{th} byte of the nn^{th} look-up table for the first band. This field supports only integer band data (PVTYPE = INT).
.....	
IREPBANDnn	This field shall contain a valid indicator of the interpretation of the nn^{th} band. Valid values are R, G, and B when IREP contains RGB; the band number is a positive integer when IREP contains MULTI. In all other cases, the use of this field is user-defined. However, its purpose is to provide the significance of the first band of the image with regard to the general image type as recorded in IREP.

TABLE IV. NITF image subheader fields - Continued.

ISUBCATnn ISUBCATnn	The use of this field is user-defined. Its purpose is to provide the significance of the nn^{th} band of the image with regard to the specific category, ICAT, of the overall image. An example would be the wavelength of IR imagery.
IFCnn	This field shall contain the value N (to mean none). Other values are reserved for future use.
IMFLTnn	This field is reserved for future use. It shall be filled with blanks.
NLUTSnn	This field shall contain the number of look-up tables associated with the nn^{th} band of the image. Use of the look-up tables is user defined in all cases after the first band.
NELUTnn	This field shall contain the number of entries in each of the look-up tables for the nn^{th} band of data. this field shall be omitted if the value in NLUTSnn is zero.
LUTDnn1	This field shall be omitted if the nn^{th} Band Number of LUTs is zero. Otherwise, this field shall contain the data defining the 1^{st} look-up table for the nn^{th} image band. Each entry in the look-up table is composed of one byte, ordered from most significant bit to least significant bit representing a value from 0 to 255. To use the look-up table, for each integer k, $0 \leq k \leq$NELUTnn-1, the pixel value k in the nn^{th} image band shall be mapped to the value of the k^{th} byte of the look-up table. This field supports only integer band data (PVTYPE = INT).
.....	
LUTDnnmm	This field shall be omitted if the nn^{th} Band Number of LUTs is zero. Otherwise, this field shall contain the data defining the mm^{th} look-up table for the nn^{th} image band. Each entry in the look-up table is composed of one byte, ordered from most significant bit to least significant bit representing a value from 0 to 255. To use the look-up table, for each integer k, $0 \leq k \leq$NELUTnn-1, the pixel value k in the nn^{th} image band shall be mapped to the value of the k^{th} byte of the look-up table. This field supports only integer band data (PVTYPE = INT).
ISYNC	This field shall contain "0" or "4", which indicates if a synchronization code has been provided for uncompressed or ARIDPCM compressed data. This field shall be set to "0" for C1 or C3 compression. (C1 and C3 have their own internal mechanism for resynchronization.) For C2 and uncompressed data, a value of "0" indicates that no code is inserted. A value of "4" indicates that a byte aligned 32-bit integer, encoded MSB to LSB, has been inserted. The value is the row number of the next row starting at zero and incrementing by eight for ARIDPCM and one for uncompressed. This code will provide a reference point for resynchronization of the image display in environments where the communications system cannot be expected to provide error free data, such as broadcast transmissions.

TABLE IV. NITF image subheader fields - Continued.

IMODE	This field shall contain an indicator of whether the image bands are stored in the file sequentially or interleaved (by block or pixel). Valid values are B, P, and S. The significance of the IMODE value must be interpreted with the knowledge of whether the image is JPEG compressed (IC=C3 or M3), VQ compressed (IC=C4 or M4), or uncompressed (IC=NC or NM). When IC=C1 or C0, the use of IMODE is undefined. The interpretation of these values of IMODE for this case is specified in Paragraph 5.2.3.3 of the NITFS document, MIL-STD-188-198A. For the uncompressed case: The value S means band Sequential, where all blocks for the first band are followed by all blocks for the second band, and so on: [(block1, band1), (block2, band1), ... (blockM, band1)], [block1, band2), (block2, band2), ... (blockM, band2)], ... [(block1, bandN), (block2, bandN), ... (blockM, bandN)]. The values B and P indicate variations on block sequential where all data from all bands for the first block is followed by all data from all bands for the second block, and so on. The variations are in the way the bands are organized within each block. B means band interleaved by Block. This means that within each block, the bands follow one another: [(block1, band1), (block1, band2), ... (block1, bandN)], [block2, band1), (block2, band2), ... (block2, bandN)], ... [(blockM, band1), (blockM, band2), ... (blockM, bandN)]. P means band interleaved by Pixel within each block: such as, for each block, one after the other, the full pixel vector (all band values) appears for every pixel in the block, one pixel after another, the block column index varying faster than the block row index. If the NBANDS field is 1, the cases B and S coincide. In this case, this field shall contain B. If the Number of Blocks is 1(NBPR = NBPC = 1), this field shall contain B for non-interleaved by pixel, and P for interleaved by pixel. The value S is only valid for images with multiple blocks and multiple bands. For the JPEG-compressed case: The presence of B, P, or S implies specific ordering of data within the JPEG image data representation. The interpretation of these values of IMODE for this case is specified in Paragraph 5.2.3.3 of the NITFS document, MIL-STD-188-198A. For the Vector Quantization compressed case: VQ compressed images are normally either RGB with a color look-up table, or monochromatic. In either case, the image is single band, and the IMODE field is undefined. However, it is possible to have a multiband VQ compressed image in band sequential, band interleaved by block, or band interleaved by pixel format.

TABLE IV. NITF image subheader fields - Continued.

NBPR	This field shall contain the number of image blocks in a row of blocks (see 5.5.1.2) in the horizontal direction. If the image consists of only a single block, this field shall contain the value one.
NBPC	This field shall contain the number of image blocks in a column of blocks (see 5.5.1.2) in the vertical direction. If the image consists of only a single block, this field shall contain the value one.
NPPBH	This field shall contain the number of pixels horizontally in each block of the image. It shall be the case that NBPR*NPPBH≥NCOLS.
NPPBV	This field shall contain the number of pixels vertically in each block of the image. It shall be the case that NBPC*NPPBV≥NROWS.
NBPP	If IC contains "NC", "NM", "C4", or "M4" this field shall contain the number of storage bits used for the value from each component of a pixel vector. The value in this field always shall be greater than or equal to Actual Bits Per Pixel (ABPP). For example, if 11-bit pixels are stored in 16 bits, this field shall contain 16 and Actual Bits Per Pixel shall contain 11. If IC = "C3," this field shall contain the value 8 or the value 12. If IC = "C1," this field shall contain the value 1, and if IC = "C2," this field shall contain the value 8 or the value 11.
IDLVL	This field shall contain a valid value that indicates the graphic display level of the image relative to other displayed file components in a composite display. The valid values are 001 to 999. The display level of each displayable file component (image, label, or symbol) within a file shall be unique; that is, each number from 001 to 999 is the display level of, at most, one item. The meaning of display level is fully discussed in 5.3.3. The image, symbol, or label component in the file having the minimum display level shall have attachment level zero. (ILOC, SLOC, and LLOC field descriptions).
IALVL	This field shall contain a valid value that indicates the attachment level of the image. Valid values for this field are 0, and the display level value of any other image, symbol, or label in the file. The meaning of attachment value is fully discussed in 5.3.4. The image, symbol, or label component in the file having the minimum display level shall have attachment level zero (ILOC, SLOC, and LLOC field descriptions).

TABLE IV. NITF image subheader fields - Continued.

ILOC	The image location is specified by specifying the location of the first pixel of the first line of the image. This field shall contain the image location represented as rrrrrccccc, where rrrrr and ccccc are the row and column offset from the ILOC, SLOC, or LLOC value of the item to which the image is attached. A row or column value of 00000 indicates no offset. Positive row and column values indicate offsets down and to the right and range from 00001 to 99999, while negative row and column values indicate offsets up and to the left and must be within the range -0001 to -9999. The coordinate system used to express ILOC, SLOC, and LLOC fields shall be common for all images, labels, and symbols in the file having attachment level zero. The location in this common coordinate system of all displayable graphic components can be computed from the offsets given in the ILOC, SLOC, and LLOC fields.
IMAG	This field shall contain the magnification (or reduction) factor of the image relative to the original source image. Decimal values are used to indicate magnification, and decimal fraction values indicate reduction. For example, 2.3 indicates the original image has been magnified by a factor of 2.3, while 0.5 indicates the original image has been reduced by a factor of 2. The default value is 1.0, indicating no magnification or reduction. In addition, the following values shall be used for reductions that are reciprocals of nonnegative powers of two: /2 (for 1/2), /4 (for 1/4), /8 (for 1/8), /16 (for 1/16), /32 (for 1/32), /64 (for 1/64), /128 (for (1/128).
UDIDL	This field shall contain the length in bytes of the sum of the following two fields (UDOFL + UDID). This length is three plus the sum of the lengths of all the registered tagged record extensions (see 5.10) appearing in the UDID field. A value of zero shall mean that no registered tagged record extensions are included in the header. If a registered tagged record extension is too long to fit in the UDID field, it shall be put in a data extension segment (see 5.9).
UDOFL	If present, this field shall contain "000" if the tagged record extensions in UDID do not overflow into a DES, or shall contain the sequence number of the DES into which they do overflow. This field shall be omitted if the field UDIDL contains zero.
UDID	If present, this field shall contain user defined registered tagged record extensions (see 5.9). The length of this field shall be the length specified by the field UDIDL, less than the length (3) of UDOFL. Registered tagged record extensions in this field for an image shall contain information pertaining specifically to the image. Registered tagged record extensions shall appear one after the other with no intervening bytes. The first byte of this field shall be the first byte of the first registered tagged record extension appearing in the field. The last byte of this field shall be the last byte of the last registered tagged record extension to appear in the field. This field shall be omitted if the field UDIDL contains zero.

TABLE IV. NITF image subheader fields - Continued.

IXSHDL	This field shall contain the length in bytes of the sum of the following two fields (IXSOFL + IXSHD). The length is three plus sum of the lengths of all the controlled tagged record extensions (see 5.9) appearing in the IXSHD field. A value of zero shall mean that no controlled tagged record extensions are included in the image subheader. If a controlled tagged record extension is too long to fit in the IXSHD field, it shall be put in a data extension segment (see 5.9).
IXSOFL	If present, this field shall contain "000" if the tagged record extensions in IXSHD do not overflow into a DES, or shall contain the sequence number in the file of the DES into which they do overflow. This field shall be omitted if the field IXSHDL contains zero.
IXSHD	If present, this field shall contain controlled tagged record extensions (see 5.9) approved and under configuration management by the NTB. The length of this field shall be the length specified by the field IXSHDL, less the length (3) of IXSOFL. Controlled tagged record extensions in this field for an image shall contain information pertaining specifically to the image. Controlled tagged record extensions shall appear one after the other in this field with no intervening bytes. The first byte of this field shall be the first byte of the first controlled tagged record extension appearing in the field. The last byte of this field shall be the last byte of the last controlled tagged record extension to appear in the field. This field shall be omitted if the field IXSHDL contains zero.

5.4.1 Field structure and default values. The NITF uses character counts to delimit header fields, as opposed to special end-of-field characters or codes or direct addressing. These counts are provided in the tables detailing the NITF header and subheader field specifications. All data in fields designated "Alphanumeric" shall be left justified and padded to the right boundary with spaces (ASCII 32, decimal). All data in numeric fields shall be right justified and padded to the left boundary with leading zeroes. The standard default value shall be zero for numeric fields and spaces for alphanumeric fields. For a few fields, a different default may be indicated in the field description. In this case, the field description shall take precedence. All header and subheader fields contained in an NITF file shall contain either valid data (that is, data in accordance with the restrictions specified for the contents of the field in this document) or the specified default value.

5.4.2 Field types. The NITF file header and various subheaders have three types of fields: Required (R), Optional (O), and Conditional (C). A Required field shall be present and shall contain valid data. An Optional field shall be present, but may contain either valid data or the default value as specified in 5.4.1. A conditional field may or may not be present depending on the value of one or more preceding (required) fields. If a conditional field is present, it shall contain valid data. When a field is conditional, its description identifies what conditions and which preceding field or fields are used to determine whether or not to include it in the file. For example, in the NITF header, if the Number of Images (NUMI) field contains the value of 2, the fields LISH001, LI001, LISH002, and LI002 will be present and must be filled with valid data. However, if the NUMI contains a zero, the subheader length and image length fields are omitted.

5.5 <u>The image data type</u>. For the NITF, the image data type encompasses multispectral imagery and images intended to be displayed as monochrome (shades of gray), color-mapped, or true color. That is, an image may include multiple data bands and color look-up tables (LUTs), the latter within its header fields. True color images (three band) may be specified to be interpreted using either the RGB (Red, Green, Blue) or the YCbCr (Y = Brightness of signal, Cb = Chrominance (blue), Cr = Chrominance (red)) color system.

5.5.1 <u>Image model</u>. For the NITF, an image is a rectangular array of pixels indexed by row and column. A pixel is an n-vector of pixel values. The i^{th} entry of the pixel (vector) is the pixel value for the i^{th} band of the image. Therefore, the i^{th} band of the image is the rectangular array of i^{th} entries from the pixel vectors. For an image I with R rows and C columns, the coordinates of the image pixel located in the c^{th} column of the r^{th} row shall be denoted by an ordered pair (r,c), $0 \le r < R, 0 \le c < C$, where the first number, r, indicates the row and the second number, c, indicates the column in the image array. This notation is standard for addressing arrays and matrices. The pixel located at (r,c) is denoted by I(r,c). For example, a typical 24-bit RGB image is an array of R rows and C columns, where each pair (r,c), $0 \le r < R, 0 \le c < C$, identifies a pixel I(r,c) consisting of three single byte values (a three-vector) corresponding to the red, green, and blue components. The image has three bands, each consisting of a R-by-C array of single byte values. One band comprises all the red, one band comprises all the green, and the third band comprises all the blue pixel components. Specifically, the value at position r,c in the green band, for example, contains the green byte from the pixel I(r,c) three-vector at position r,c in the image.

5.5.1.1 <u>Display of NITF images</u>. When an image I with R rows and C columns is displayed, a mapping is accomplished from the stored image array I to a rectangular array S of physical picture elements, for example a Cathode Ray Tube (CRT) display. This mapping will be called the display mapping. Usually, the resulting display has an identified top, bottom, and left and right side. In a particular application, the display mapping may be defined explicitly. However, lacking this, an image stored in a NITF file shall be interpreted so that pixel I(0,0) is at the upper left corner, and pixel I(R-1,C-1) is at the lower right corner. The r^{th} row of the image array I shall form the r^{th} row of the display, counting from the top, $0<=r<R$. Within the r^{th} row, the pixels shall appear beginning on the left with I(r,0) and proceeding from left to right with I(r,1), I(r,2), and so on, ending with I(r, C-1). Figure 6 illustrates the display mapping just described. The relationship of the pixels I(r,c) in the image array to up, down, left and right implicit in this diagram is used freely in later descriptions to simplify exposition.

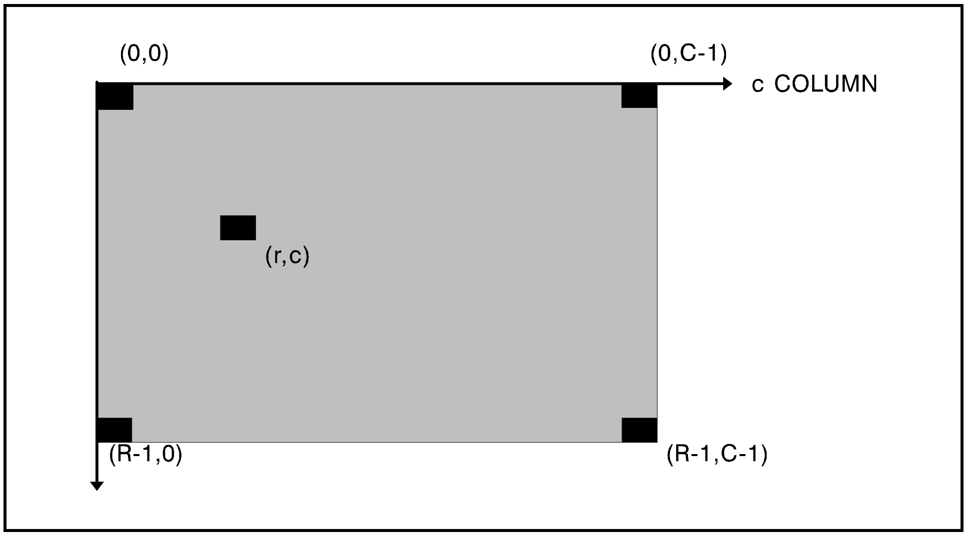


FIGURE 6. Image coordinate system.

5.5.1.2 Blocked images. The concept of a subimage is introduced here to help discuss blocked images, which extend the image model for NITF presented above to support the representation of an image in terms of an orderly set of subimages (or subarrays) called blocks. Let I be an image with R rows and C columns. Let r0, RB, c0, and CB be integers that satisfy $0 \leq r0 < R$, $0 \leq c0 < C$, $0 < RB \leq R - r0$ and $0 < CB \leq C - c0$. The image B with pixels B(r',c') defined by

$$B(r',c') = I(r0 + r',c0 = c')$$

for $0 \leq r' < RB$ and $0 \leq c' < CB$, is a subimage of I having RB rows and CB columns. Figure 7 illustrates the subimage concept. The idea behind a blocked image is to view it as comprising a rectangular array of uniform, adjoining subimages called blocks. A rectangular tiled floor is a suitable analogy. Regard the overall floor cover as the image and each individual tile as a block. To make this more precise, let I be an image of R rows and C columns, and let the Number of Pixels Per Block Horizontal (NBBPH), (that is, the number of columns of each block) and the Number of Pixels Per Block Vertical (NBBPV), (that is, the number of rows in each block) be positive integers that satisfy NPPBH≤C and NPPBV≤R. If R is an integral multiple of NPPBV and C is an integral multiple of NPPBH, then I may be viewed as an array B of subimages each of having NPPBV rows and NPPBH columns. Specifically, for each pair of non-negative integers r,c for which (c+1)*NPPBH≤C, and (r+1)*NPPBV≤R, a subimage $B_{r,c}$ may be formed from I by letting r0 = r*NPPBV, RB = NPPBV and c0 = c*NPPBH, CB = NPPBH. These subimages $B_{r,c}$ are called blocks. The block $B_{r,c}$ is in the r^{th} row of blocks and the c^{th} column of blocks. The number of columns of blocks is the integer, C/NPPBH, and the number of rows of blocks is the integer R/NPPBV. The relation of image blocks to image rows and columns is depicted on figure 8a using the NITF display convention described in 5.5.2. If the number of rows in an image is not initially an integer multiple of NPPBV, or if the number of columns is not an integer multiple of NPPBH, an application that uses the blocked image construct in NITF shall "pad" the image to an appropriate number of rows and columns so the divisibility condition is met by adding rows to the bottom and/or columns to the right side of the image, as viewed. The result is that a blocked image may have a subimage comprising the "significant" pixels, that is those from the original image. The remaining pixels are "pad." Figure 8b illustrates this situation.

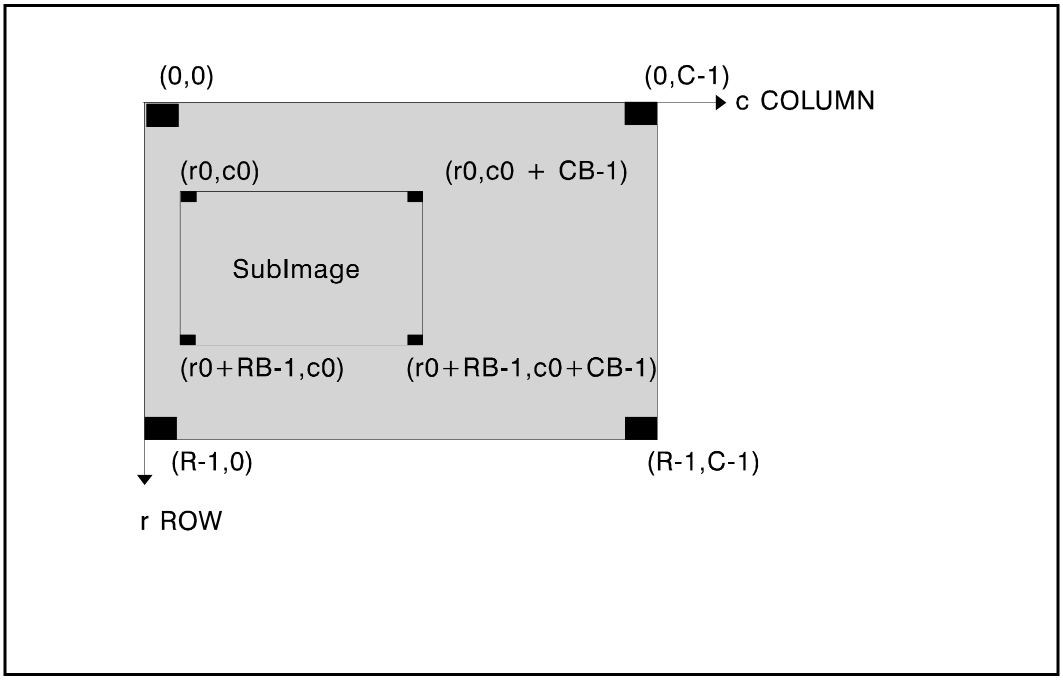


FIGURE 7. Subimage with RB rows and CB columns.

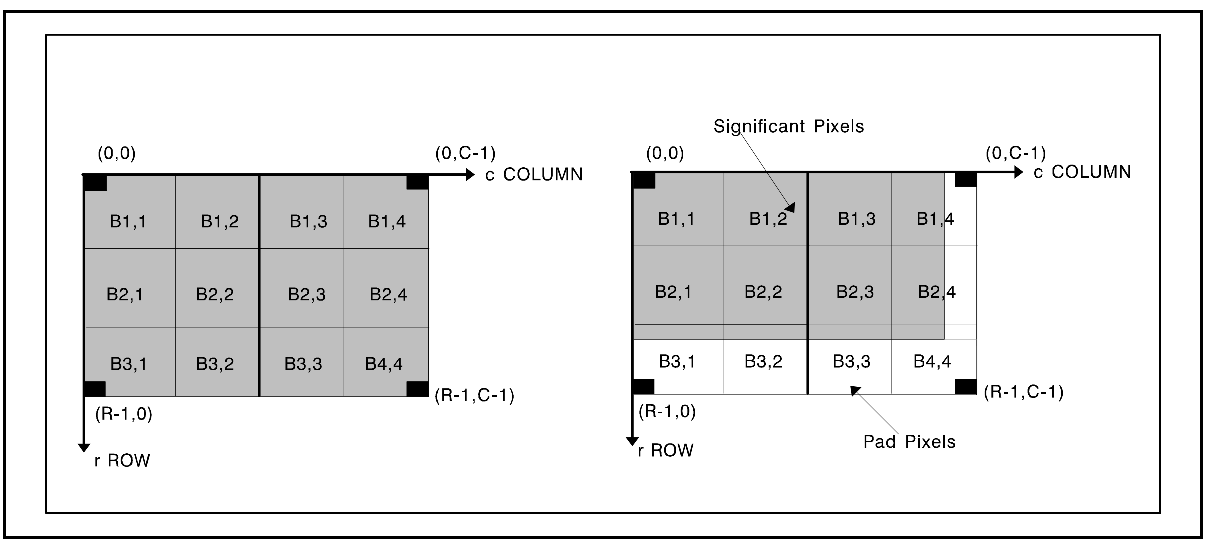


FIGURE 8a. A blocked image. FIGURE 8b. A blocked padded image.

5.5.1.3 Blocked image masking. In some instances, a blocked image may have a considerable number of empty blocks. This might occur when a rectangular image is not north aligned, but has been rotated to a north up orientation (see figure 8c). In this case, it is sometimes useful to not record or transmit empty blocks. however, if empty blocks are not recorded/transmitted, the image loses its logical structure as an image with n x m blocks. In order to preclude the loss of logical structure, and to allow the exclusion of empty blocks, a Mask Subheader structure has been defined. The Mask Subheader is defined in TABLE IV(A) and TABLE IV(B). The mask identifies the location of non-empty blocks, and tags empty blocks so that the using application can reconstruct the image correctly. In figure 8c, the recording order would be B1,1; B1,2; B1,3; B2,1; B2,2; B2,3; B2,4; B3,1; B3,2; B3,3; B3,4; B4,2; B4,3; B4,4. The blocked image mask would identify the locations of the recorded image blocks. If the image is band sequential (IMODE=S), there will be multiple block image masks--one for the image bands, with each mask containing NBPR x NBPC records. Block image masks can be used in conjunction with a transparent pixel mask, as described below. A block image mask may also be used to provide random access within the blocked image data for large images even if all blocks are recorded.

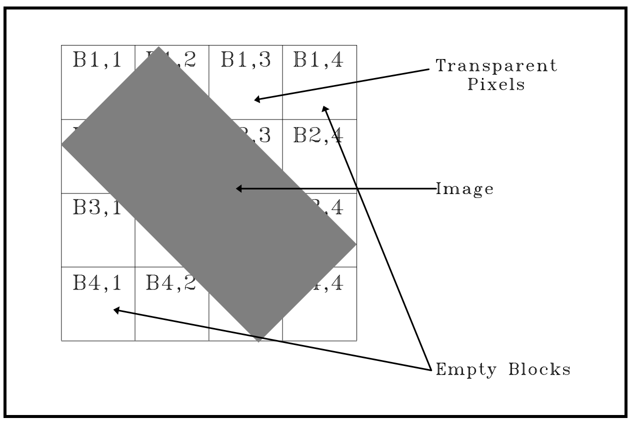


FIGURE 8c. A blocked,padded image with empty blocks.

5.5.1.4 Transparent pixel masking. In addition to empty image blocks, Figure 8c also demonstrates that a significant number of transparent pixels may be needed to "pad" an image to the nearest block boundary. In conjunction with the blocked image mask described in detail in TABLE IV(A). In the example in Figure 8c, the locations of image blocks 0, 1, 2, 4, 6, 7, 8, 9, 11, 13, 14, and 15 would be recorded, indicating that blocks B1,1; B1,2; B1,3; B2,1; B2,3; B2,4; B3,1; B3,2; B3,4; B4,2; B4,3; and B4,4 have transparent pixels. B1,4; B2,2; B3,3; and B4,1 do not have transparent pixels because B1,4 and B4,1 are empty and B2,2 and B3,3 are full image blocks. If the image is band sequential (IMODE=S), there will be pixel masks that will be arranged in the same order as the image bands, with each mask containing NBPR x NBPC records. The output pixel code which represents transparent pixels is identified within the Image Data Mask Subheader by the Transparent Output Pixel Code field (TPXCD). The length in bits of this code is identified in the Transparent Output Pixel Code Length field (TPXCDLNTH). Although this length is given in bits, the actual TPXCD value is stored in an integral number of bytes. When the number of bits used by the code is less than the number available in the TPXCD field (for example, a 12 bit code stored in two bytes), then the code will be justified in accordance with the PJUST field in the Image

Subheader. When an application identifies transparent pixels, it may replace them with a user defined value (for example, alight blue background) at the time of presentation. The application may also choose to ignore transparent pixels in histogram generation. In any case, transparent pixels are not valid data, and should not be used for interpretation or exploitation.

5.5.1.5 Image data mask subheader. The image data mask subheader is a conditional data structure included in the image data stream for masked images (IC values NM, M0, M3, and M4). The image data mask subheader is not recorded for non-masked images (IC values NC, C0, C1, C2, C3, and C4) and identical to that of non-masked images except for the following: the first byte of the image data is offset from the beginning of the image data area by the length of the image data mask subheader; and empty image blocks are not recorded/transmitted in the image data area. If the image is band sequential (IMODE=S), there will be multiple block image and/or transparent pixel masks--one for each band. All block image masks will be recorded first, followed by all transparent pixel masks. Since the image data mask subheaders are in the image area, the data recorded/transmitted there are binary. The structure of the image data mask subheader is defined in detail in TABLE IV(A) and TABLE IV(B).

TABLE IV(A). NITF image data mask subheader.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
IMDATOFF	Blocked Image Data Offset	4	Unsigned Integer: 0 to 2^{32} - 1	C
BMRLNTH	Block Mask Record Length	2	Unsigned Integer; 0=No Block mask; 4=Block mask present	C
TMRLNTH	Transparent Pixel Mask Record Length	2	Unsigned Integer; 0=No transparent pixel mask; 4=Transparent pixel mask Present	C
TPXCDLNTH	Transparent Output Pixel Code Length	2	Unsigned Integer; 0=No transparent pixels; or Transparent pixel code length in bits	C

TABLE IV(A). NITF image data mask subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
TPXCD	Transparent Output Pixel Code	*	Unsigned Integer; 0 to 2^n - 1 where n=TPXCDLNTH	C
BMR0BND1	Block Mask Record 0. Band 1	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block 0 of band 1 (usually 0); OxFFFFFFFF if the block is not recorded	C
....				
BMRnnBND1	Block Mask Record nn, Band 1	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block nn of band 1; OxFFFFFFFF if the block is not recorded	C
....				
BMRnnBNDmm	Block Mask Record nn, Band mm	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block nn of band mm; OxFFFFFFFF if the block is not recorded	C

TABLE IV(A). NITF image data mask subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
TMR0BND1	Transparent Pixel Mask Record 0, Band 1	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block 0 of band 1; OxFFFFFFFF if the block does not contain transparent pixels	C
TMRnnBND1	Transparent Pixel Mask Record nn, Band 1	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block nn of band 1; OxFFFFFFFF if the block does not contain transparent pixels	C
....				
TMR0BNDmm	Transparent Pixel Mask Record 0, Band mm	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block 0 of band mm; OxFFFFFFFF if the block does not contain transparent pixels	C
....				

TABLE IV(A). NITF image data mask subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
TMRnnBNDmm	Transparent Pixel Mask Record nn, Band mm	4	Unsigned Integer; Offset in bytes from the beginning of Blocked Image Data to the first byte of block nn of band mm; OxFFFFFFFF if the block does not contain transparent pixels	C

* The length of the TPXCD field is next highest number of bytes which can contain the number of bits identified in the TPXCDLNTH field. For example, a TPXCDLNTH value of 12 would be stored in a TPXCD field of two bytes.

TABLE IV(B). NITF image data mask subheader fields

IMDATOFF	This field is included if the IC value equals NM, M0, M3, or M4. It identifies the offset from the beginning of the Image Data Mask Subheader to the first byte of the blocked image data. This offset, when used in combination with the offsets provided in the BMR fields, can provide random access to any recorded image block in any image band.
BMRLNTH	This field is included if the IC value equals NM, M0, M3, or M4. It identifies the length of each Block Mask Record in bytes. The total length of the Block Mask Records is equal to BMRLNTH x NBPR x NBPC x NBANDS. If all of the image blocks are recorded, this value is set to 0, and the conditional BMR fields are not recorded/transmitted. If this field is present, but coded as 0, then a transparent pixel mask is included.

TABLE IV(B). NITF image data mask subheader fields - Continued.

TMRLNTH	This field is included if the IC value equals NM, M0, M3, or M4. It identifies the length of each Transparent Pixel Mask Record in bytes. The total length of the Transparent Pixel Mask Records is equal to TMRLNTH x NBPR x NBPC x NBANDS. If none of the image blocks contain transparent pixels, this value is set to 0, and the conditional TMR fields are not recorded/transmitted. For IC value of M3, the value is set to 0. If this field is present, but coded as 0, then a Block Mask is included.
TPXCDLNTH	This field is included if the IC value equals NM, M0, M3, or M4. It identifies the length in bits of the Transparent Output Pixel Code. If coded as 0, then no transparent pixels are present, and the TPXCD field is not recorded. For IC value of M3, the value is set to 0.
TPXCD	This field is included if the IC value equals NM, M0, or M4, and TPXCDLNTH is not 0. It contains the output pixel code that represents a transparent pixel in the image. This value is unique within the image, and allows the user to identify transparent pixels. The transparent pixel output code length is determined by TPXCDLNTH, but the value is stored in two bytes. If the number of bits used by TPXCD is less than the number of bits available for storage, the value shall be justified in accordance with the PJUST field in the image subheader.
BMR0BND1	This field shall contain the first Block Mask Record of band 1. It is recorded/transmitted only if the BMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the Blocked Image Data to the first byte of block 0 of band 1 (this value should be 0) if block 0 is recorded/transmitted, or 0xFFFFFFFF if block 0 of band 1 is not recorded/transmitted in the image data.

TABLE IV(B). NITF image data mask subheader fields - Continued.

BMR0BND1	This field shall contain the nnth Block Mask Record of band 1. It is recorded/transmitted only if the BMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the blocked Image Data to the first byte of block nn of band 1 if block nn is recorded/transmitted, or 0xFFFFFFFF if block nn of band 1 is not recorded/transmitted in the image data. The number of BMR records for this band is NBPR x NBPC.
BMR0BNDmm	This field shall contain the first Block Mask Record of band mm. It is recorded/transmitted only if the BMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the Blocked Image Data to the first byte of block 0 of band mm if block 0 is recorded/transmitted, or 0xFFFFFFFF if block 0 of band mm is not recorded/transmitted in the image data.
BMRnnBNDmm	This field shall contain the nnth Block Mask Record of band mm. It is recorded/transmitted only if the BMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the Blocked Image Data to the first byte of block nn of band mm if block nn of band mm is not recorded/transmitted in the image data. The number of BMR records for this band is NBPR x NBPC.
TMR0BND1	This field shall contain the first Transparent Pixel Mask Record for band 1. It is recorded/transmitted only if the TMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the blocked Image Data to the first byte of block 0 of and 1 if block- contains transparent pixels, or 0xFFFFFFFF to indicate that this block does not contain transparent pixels.

TABLE IV(B). NITF image data mask subheader fields - Continued.

TMRnnBND1	This field shall contain the nnth Transparent Pixel Mask Record for band 1. It is recorded/transmitted only if the TMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the Blocked Image Data to the first byte of block nn of band 1 if block nn contains transparent pixels, or 0xFFFFFFFF to indicate that this block does not contain transparent pixels. The number of TMR records for band 1 is NBPR x NBPC.
TMR0BNDmm	This field shall contain the first Transparent Pixel Mask Record for band mm. It is recorded/transmitted only if the TMRLNTH field is not 0. The field shall contain and offset in bytes form the beginning of the Blocked Image Data to the first byte of block 0 of band mm if block 0 contains transparent pixels, or 0xFFFFFFFF to indicate that this block does not contain transparent pixels.
TMRnnBNDm	This field shall contain the nnth Transparent Pixel Mask Record for band mm. It is recorded/transmitted only if the TMRLNTH field is not 0. The field shall contain an offset in bytes from the beginning of the Blocked Image Data to the first byte of block nn of band mm if block nn contains transparent pixels, or 0xFFFFFFFF to indicate that this block does not contain transparent pixels. The number of TMR records for band mm is NBPR x NBPC.

5.5.2 NITF image information fields. In the NITF, the information describing an image is represented in a series of adjacent fields grouped into the image subheader followed by the image data. The field containing the image data is called the image data field. The image data field shall follow immediately the last field of the corresponding image subheader with no intervening special characters to designate the beginning of the image data field. Similarly, the image subheader of the first image shall follow immediately the last byte of data of the last field in the NITF header, and the image subheader of successive images shall follow immediately the last byte of the image data field of the preceding image.

5.5.2.1 Image subheader fields. The data in the image subheader fields are ASCII-encoded character data. They provide information about the image source, its identification, and characteristics needed to display and interpret it properly. The image subheader field definitions are detailed in table III and table IV.

5.5.2.2 Image data field format. Image data may be stored in an NITF file in either compressed or uncompressed form.

5.5.2.2.1 Compressed image data format. The format of the image data placed in the image data field after compression is provided with the description of the NITFS image compression algorithms in MIL-STD-188-196, MIL-STD-188-197A, MIL-STD-188-198A, and MIL-STD-188-199. Also found in these references are the conditions the data must meet before a given compression method can be applied meaningfully.

5.5.2.2.2 Uncompressed image data format. The order in which pixel values of a single band image are stored is fixed. When an image has more than one band, several options are available for the order in which pixel values are stored. The option used is indicated by the IMODE field in the image subheader. The following subparagraphs describe the possibilities within this format. In describing the encoding of image data, the NITF display convention is invoked freely for ease of expression. Let the image to be encoded be denoted by I, and assume I has R rows and C columns. Let I have n bands; that is, each pixel is an n-vector, the i^{th} value of which is the value for that pixel location of the i^{th} band of the image. Let N denote the number of bits-per-pixel-per-band. Thus, there are n*N bit-per-pixel. Let I be blocked with H blocks per row and V blocks per column. Note that special cases such as single band images and single block images are included in this general image by setting n=1, and H=V=1, respectively.

5.5.2.2.2.1 Single band image uncompressed data format. For single band images, n=1, and there is only one order for storing pixels. The field IMODE in the image subheader shall be set to B for this case. The blocks (one or more) shall be stored, one after the other starting with the upper left block and proceeding first left to right across rows of blocks, one row of blocks after the other, top to bottom. Image data within each block shall be encoded as one continuous bit stream, one pixel value after another, beginning with the N bits of the upper left corner pixel, I(0,0), followed by the N bits of I(0,1) and so on until all pixels from the first row in the block are encoded. These shall be followed immediately by the N bits of data for pixel I(1,0) continuing from left to right along each row, one row after another from the top of the block to the bottom. The last byte of each block's data is zero-filled to the next byte boundary, but all other byte boundaries within the block are ignored. See the field Pixel Value Type (PVTYPE) description in table IV for the specification of the bit representation of pixel values.

5.5.2.2.2.2 Multiple band image uncompressed data format. For multiple band images, there are three orders for storing pixels.

5.5.2.2.2.2.1 Band sequential. The first case is "band sequential", in which each band is stored contiguously, starting with the first band, one after the other, until the last band is stored. Within each band the data shall be encoded as if it were a single band image with one or more blocks (see 5.5.2.2.2.1). The field IMODE in the image subheader shall be set to S for this case.

5.5.2.2.2.2.2 Band interleaved by pixel. The ordering mechanism for this case stores the pixels in a block sequential order in which each block is stored contiguously, starting with the upper left block and proceeding first left to right across rows of blocks, one row of blocks after the other, top to bottom. Within each block, the multiple band image data can be stored in one of two ways, either interleaved by pixel or by block. For "band interleaved by pixel" the n*N bits of the entire pixel vector are stored pixel-by-pixel in the same left to right, top to bottom pixel order as described in 5.5.2.2.2.1. The n*N bits for a single pixel are stored successively in this order: the N bits of the first band followed by the N bits of the second band and, so forth, ending with the N bits of the last band. Each block shall be zero-filled to the byte boundary. The field IMODE in the image subheader shall be set to P for this storage option. See the field Pixel Value Type (PVTYPE) description in table IV for the specification of the bit representation of pixel values for each band.

5.5.2.2.2.2.3 Band interleaved by block. The ordering mechanism for this case stores the pixels in a block sequential order where each block is stored contiguously, starting with upper left block and proceeding first left to right across rows of blocks, one row of blocks after the other, top to bottom. Within each block, the multiple band image data can be stored in one of two ways, either interleaved by pixel or by block. For "band interleaved by block" the data from each band is stored starting with the first band, one after the other until the last band is stored. Each block shall be zero-filled to the next byte boundary. The field IMODE in the image subheader shall be set to B for this storage option. This case is only valid for images with more than one block. (For single block images, this case collapses to the "band sequential" case where IMODE is set to S.) See the field Pixel Value Type (PVTYPE) description in table IV for the specification of the bit representation of pixel values for each band.

TABLE V. Security control markings.

CODEWORD	DIGRAPH
NOCONTRACT	NC
ORCON	OR
PROPIN	PI
WNINTEL	WI
LIMDIS	DS
ATOMAL	AL
COSMIC	CS
CNWDI	CN
CRYPTO	CR
FOUO	FO
FORM REST DATA	RD
SIOP	SH
SIOP/ESI	SE
COPYRIGHT	PX
EFTO	TX
LIM OFF USE (UNCLA)	LU
NONCOMPARTMENT	NT
NOFORN	NF
PERSONAL DATA	IN
SAO	SA
SAO-1	SL
SAO-2	HA
SAO-3	HB
SAO-SI-2	SK
SAO-SI-3	HC
SAO-SI-4	HD
SPECIAL CONTROL	SC
SPECIAL INTEL	SI
SI-1	SN
WARNING NOTICE- SEC CLAS IS BASED ON THE FACT OF EXISTENCE AND AVAIL OF THIS GRAPHICS	WN

5.6 Symbol data type. The symbol data field is used in the NITF to store a two-dimensional graphical symbol represented as a bit-map, as an NITF-defined object, or as a Computer Graphics Metafile (CGM). A symbol may be black and white, gray scale, or color. Examples of symbols are circles, ellipses, rectangles, arrows, lines, triangles, logos, unit designators, object designators (ships, aircraft), and special characters. A symbol is stored as a distinct unit in the NITF file allowing it to be manipulated and displayed nondestructively relative to the images, labels, and other symbols in the file.

5.6.1 Symbol subheader. Each symbol has a symbol subheader immediately preceding it in the NITF file. The symbol subheader is used to identify and supply the information necessary to display the symbol as intended by the file builder. The format for the symbol subheader is shown in table VI. Descriptions of the subheader parameters follow in table VII.

TABLE VI. NITF symbol subheader.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
SY	File Part Type	2	SY	R
SID	Symbol ID	10	Alphanumeric	R
SNAME	Symbol Name	20	Alphanumeric	O
SSCLAS	Symbol Security Classification	1	T, S, C, R, or U	R
SSCODE	Symbol Codewords	40	Alphanumeric	O
SSCTLH	Symbol Control and Handling	40	Alphanumeric	O
SSREL	Symbol Releasing Instructions	40	Alphanumeric	O
SSCAUT	Symbol Classification Authority	20	Alphanumeric	O
SSCTLN	Symbol Security Control Number	20	Alphanumeric	O
SSDWNG	Symbol Security Downgrade	6	Alphanumeric	O
SSDEVT	Symbol Downgrading Event	40	Alphanumeric	C
ENCRYP	Encryption	1	0=Not Encrypted 1=Encrypted	R
STYPE	Symbol Type	1	B for Bit-mapped; C for CGM; O for Object	R
NLIPS	Number of Lines Per Symbol	4	0-9999	R
NPIXPL	Number of Pixels Per Line	4	0-9999	R
NWDTH	Line Width	4	0-9999	R
NBPP	Number of Bits Per Pixel	1	0-8	R
SDLVL	Display Level	3	1-999	R
SALVL	Attachment Level	3	0-998	R
SLOC	Symbol Location	10	rrrrrccccc	R

TABLE VI. NITF symbol subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
SLOC2	Second Symbol Location	10	rrrrrccccc	O
SCOLOR	Symbol Color	1	See Symbol Color description	R
SNUM	Symbol Number	6	Alphanumeric (table IV)	O
SROT	Symbol Rotation	3	0-359	R
NELUT	Number of LUT Entries	3	0-256	R
DLUT	Symbol LUT Data	†	Pixel values in Order	C
SXSHDL	Extended Subheader Data Length	5	0-08833	R
SXSOFL	Extended Subheader Overflow	3	0-999	C
SXSHD	Extended Subheader Data	*	Controlled Tagged Record Extensions	C

* As specified by the SHSHDL field
† For color LUT symbols, size of DLUT=3 x NELUT
For gray scale LUT symbols, size of DLUT=NELUT

TABLE VII. NITF symbol subheader fields.

SY	This field shall contain the characters SY to identify the subheader as a symbol subheader.
SID	This field shall contain a valid alphanumeric identification code associated with the symbol. The valid codes are determined by the application.
SNAME	This field shall contain an alphanumeric for the symbol.
SSCLAS	This field shall contain a valid value representing the classification level of the symbol. Valid values are: T (=Top Secret), S (=Secret), C (=Confidential), R (=Restricted), U (=Unclassified).
SSCODE	This field shall contain a valid indicator of the security compartments associated with the symbol. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, and complete codewords or project numbers. The selection of a relevant set of codewords and project numbers is application specific. If this field is all spaces, it shall imply that no codewords apply to the symbol.

TABLE VII. NITF symbol subheader fields - Continued.

SSCTLH	This field shall contain valid security handling instructions associated with the symbol. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, complete words and abbreviations of more than two characters, and phrases only if the words within the phrase are separated by hyphens. The selection of a relevant set of security handling instructions is implementation specific. If this field is all spaces, it shall imply that no symbol control and handling instructions imply.
SSREL	This field shall contain a valid list of countries and/or groups of countries to which the symbol is authorized for release. Valid items in the list are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: country codes and groupings that are digraphs in accordance with FIPS PUB 10-3. If this field is all spaces, it shall imply that no symbol release instructions imply.
SSCAUT	This field shall contain a valid identity code of the classification authority for the symbol. The code shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no symbol classification authority applies.
SSCTLN	This field shall contain a valid security control number associated with the symbol. The format of the security control number shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no symbol security control number applies.
SSDWNG	This field shall contain a valid indicator that designates the time at which a declassification or downgrading action is to take place. The valid values are (1) the calendar date in the format YYMMDD, (2) the code "999999" when the originating agency's determination is required (OADR), and (3) the code "999998" when a specific event determines at what time declassification or downgrading takes place. If this field is all spaces, it shall imply that no symbol security downgrade condition applies.
SSDEVT	If the Symbol Security Downgrade field (SSDWNG) equals "999998," this field shall be present and shall contain a valid specification of the downgrade event. If this field is present and all spaces, it shall constitute an error. Valid values for the event specification are determined by the application.
ENCRYP	This field shall contain the value zero until such time as this specification is updated to define the use of other values.

TABLE VII. NITF symbol subheader fields - Continued.

STYPE	This field shall contain a valid indicator of the representation type of the symbol. Valid values are B, C, and O. B means bit-mapped. For bit-mapped symbols, the symbol parameters are found in the symbol subheader, and the symbol data values are contained in the symbol data field immediately following the subheader. C means Computer Graphics Metafile. The symbol data contain a Computer Graphics Metafile in binary format that defines the symbol according to the specification of CGM for NITF in NITFS MIL-STD-2301. O means object. The Symbol Number (SNUM) is a reference number that indicates the specific symbol as defined in table VIII. No symbol data field if this shall be present contains O, since an object symbol only has a subheader. The currently defined objects are standard geometric shapes and annotations of sufficient simplicity that they can be implemented accurately from verbal descriptions. Future versions of the NITF will include various predefined objects such as symbols for military units, vehicles, weapons, aircraft.
NLIPS	If STYPE = B or O, this field shall contain the number of rows (lines) in the symbol image. This field shall contain zero if STYPE = C.
NPIXPL	If STYPE = B or O, this field shall contain the number of pixels in each row (line) of the symbol (equals the number of image columns in the symbol viewed as an image). This field shall contain zero if STYPE = C.
NWDTH	If STYPE = O, this field shall contain the line width for the object symbol in pixels. If this field equals the value in NLIPS, the symbol should be drawn solid (filled in). This field shall contain zero if STYPE = C or B.
NBPP	If STYPE = B, this field shall contain the number of storage bits used for the value of each pixel in the symbol. If STYPE = C, this field shall contain zero. If STYPE = O, this field shall contain the value "1".
SDLVL	This field shall contain a valid value that indicates the graphic display level of the symbol relative to other displayed file components in a composite display. The valid values are 001 to 999. The display level of each displayable file component (image, label, or symbol) within a file shall be unique; that is, each number from 001 to 999 is the display level of, at most, one item. The meaning of display level is discussed fully in 5.3.3. The symbol, image, or label component in the file having the minimum display level shall have attachment level zero (ILOC, SLOC, and LLOC field descriptions).
SALVL	This field shall contain a valid value that indicates the attachment level of the symbol. Valid values for this field are 0 and the display level value of any other image, symbol, or label in the file. The meaning of attachment value is discussed fully in 5.3.4. The symbol, image, or label component in the file having the minimum display level shall have attachment level zero (ILOC, SLOC, and LLOC field descriptions).

TABLE VII. NITF symbol subheader fields - Continued.

SLOC	The symbols location is specified by providing the location of a point bearing a particular relationship to the symbol. For a bit-mapped symbol, the point is the first pixel of the first row. For an object symbol, the point is specified in table VIII as part of each symbol's definition. For a CGM symbol, the point is defined in MIL-STD-2301. This field shall contain the symbol location represented as rrrrrccccc, where rrrrr and ccccc are the row and column offset from the ILOC, SLOC, or LLOC value of the item to which the symbol is attached. A row and column value of 00000 indicates no offset. Positive row and column values indicate offsets down and to the right and range from 00001 to 99999, while negative row and column values indicate offsets up and to the left and must be within the range -0001 to -9999. The coordinate system used to express ILOC, SLOC, and LLOC fields shall be common for all images, labels, and symbols in the file having attachment level zero. The location in this common coordinate system of all displayable graphic components can be computed from the offsets given in the ILOC, SLOC, and LLOC fields.
SLOC2	This field shall contain an ordered pair of integers defining a location in Cartesian coordinates for use with object symbols. The meaning of this location is defined in table X for object symbols. The format is rrrrrccccc, where rrrrr is the row and ccccc is the column offset from the ILOC, CLOC, or LLOC value of the item to which the symbol is attached. If the symbol is unattached (SALVL = 0), rrrrr and ccccc represent offsets from the origin of the coordinate system that is common to all images, labels, and symbols in the file having attachment level zero. rrrrr and ccccc each range from -9999 to 99999.
SCOLOR	If STYPE = B, this field shall contain a valid single character code from among the following indicating how the bit-mapped symbol shall be color-mapped. If STYPE = O, this field shall contain a valid single character code from among the following list indicating how the object symbol shall be color-mapped. The object shall be rendered using the color for the value "1". If NBPP≥1, the following are valid: C to mean "use included Color Look-Up Table" G to mean "use included Gray Scale Look-Up Table" If NBPP = 1, the following are additionally valid: N to mean "interpret 0=Black, 1-White" K to mean "interpret 0=Transparent, 1=Black" W to mean "interpret 0=Transparent, 1=White" R to mean "interpret 0=Transparent, 1=Red" O to mean "interpret 0=Transparent, 1=Orange" B to mean "interpret 0=Transparent, 1=Blue" Y to mean "interpret 0=Transparent, 1=Yellow" If STYPE = C, this field shall contain the space character (ASCII 32, decimal).

TABLE VII. NITF symbol subheader fields - Continued.

SNUM	For object symbols, this field shall contain the unique numeric identifier (values 1-18) of one of the objects defined in table X. For bit-mapped and CGM symbols, this field shall contain 000000. The field is alphanumeric to support future use of alphanumeric symbol identifiers.
SROT	When STYPE = O, this field shall contain the rotation angle of the symbol in integer degrees about its rotation point in the counterclockwise direction with respect to the nominal orientation. Nominal orientation is the orientation corresponding to SROT = 000. If STYPE = B or C, this field shall contain 000, and shall be ignored.
NELUT	When STYPE = B, this field shall contain the number of entries in the look-up table associated with the symbol. Valid values are 0, 2, 4, 8, 16, 32, 64, 128, and 256. 0 shall be interpreted to mean no LUT is present (no data in DLUT field.) This field shall contain a zero when STYPE = O or C.
DLUT	If present, this field shall contain the data defining the color look-up table for the symbol. The data format for the two types of look-up tables (gray scale and color) is described in 5.6.2.1.1 and 5.6.2.1.2. This field shall not be present (NELUT = 0) if STYPE = O or C.
SXSHDL	This field shall contain the length in bytes of the sum of the following two fields (SXSOFL + SXSHD). This length is three plus the sum of the lengths of all the controlled tagged record extensions (see 5.9) appearing in the SXSHD field. A value of zero shall mean that no controlled tagged record extensions are included in the symbol subheader. If a controlled tagged record extension is too long to fit in the SXSHD field, it shall be put in an data extension segment (see 5.9).
SXSOFL	If present, this field shall contain "000" if the tagged record extensions in SXSHD do not overflow into a DES, or shall contain the sequence number in the file of the DES into which they do overflow. This field shall be omitted if the field SXSHDL contains zero.
SXSHD	If present, this field shall contain controlled tagged record extensions (see 5.9) approved and under configuration management by the NTB. The length of this field shall be the length specified by the field SXSHDL, less the length (3) of SXSOFL. Controlled tagged record extensions in this field for a symbol shall contain information pertaining specifically to the symbol. Controlled tagged record extensions shall appear one after the other in this field with no intervening bytes. The first byte of this field shall be the first byte of the first controlled tagged record extension appearing in the field. The last byte of this field shall be the last byte of the last controlled tagged record extension to appear in the field. This field shall be omitted if the field SXSHDL contains zero.

5.6.2 Symbol data.

5.6.2.1 Bit-mapped symbols. A bit-mapped symbol is a special type of image. The image model, coordinate system, and display convention in 5.5.1.1 and 5.5.1.2 for NITF images is applicable to bit-mapped symbols. For each bit-mapped symbol in the file, the symbol data will follow the subheader immediately. Bit-mapped symbols can be regarded as images with C pixels horizontally and R pixels vertically with from one to eight bits of data for each pixel. Bit-mapped symbols either may be gray scale or color (includes black and white).

5.6.2.1.1 Bit-mapped symbol gray scale look-up tables (LUT). The gray scale to be used in displaying each pixel of a gray scale bit-mapped symbol is determined using the symbol's LUT. A LUT for a gray scale symbol shall comprise a one byte entry for each integer (the entry's index) in the range 0 to NELUT-1. The bytes of the LUT shall appear in the file one after the other without separation. The entries shall occur in the index order, the first entry corresponding to index 0, the second to index 1 and so on, the last corresponding to index NELUT-1. The display shade for a pixel in the symbol shall be determined by using the symbol pixel value as an index into the LUT. The LUT value shall correspond to the display gray shade in a way specific to the display device. However, a LUT entry value of zero shall cause display transparency (the pixel is not displayed); a value of 001 shall cause black to be displayed, and a value of 255 shall cause white to be displayed. NELUT shall be greater than the maximum pixel value in the symbol to ensure that all symbol pixels are mapped to the display device.

5.6.2.1.2 Bit-mapped symbol color look-up tables. Color bit-mapped symbols are represented using the RGB color system notation. For color symbols, each LUT entry shall be composed of three bytes; one byte each for the output color components red, green, and blue, appearing in the file in that order. There shall be a LUT entry for each integer (the entry's index) in the range 0 to NELUT-1. The three byte entries shall appear in the file in increasing index order beginning with index 0. The display color of a symbol pixel shall be determined by using the pixel value as an index into the LUT. The corresponding values for red, green, and blue shall determine the displayed color in a manner specific to the display device. However, this standard specifies the LUT values for certain predefined colors. These are presented in table VIII in binary notation for red, green, and blue components of each LUT entry. A color look-up table shall have at least two entries if the symbol is represented by one bit-per-pixel, four entries for two bits-per-pixel, and so on with a maximum of 256 entries for eight bits-per-pixel. Table IX shows a sample look-up table for a symbol with three colors represented by two bits-per-pixel.

TABLE VIII. Predefined colors.

R	G	B	MEANING
00000000	00000000	00000000	Transparent
00000001	00000001	00000001	Black
11111111	00000000	00000000	Red

TABLE VIII. Predefined colors - Continued.

R	G	B	MEANING
00000000	11111111	00000000	Green
000000000	00000000	11111111	Blue
11111010	01111101	00000000	Orange
11111010	11111010	00000000	Yellow
11111111	11111111	11111111	White

TABLE IX. Sample look-up table.

LOOK-UP TABLE DATA				
PIXEL VALUE	**R**	**G**	**B**	**MEANING**
00	00000000	00000000	00000000	Transparent
01	00000001	00000001	00000001	Black
10	11111111	00000000	00000000	Red
11	11111111	11111111	11111111	White

5.6.2.1.3 Bit-mapped symbol data representation. The data for a R row by C column bit-mapped symbol of N bits-per-pixel shall begin with the N bits-per-pixel (0,0), in order, beginning with the most significant bit (MSB) and ending with the least significant bit (LSB). Next are the N bits for each of the remaining pixels in the first row, in increasing column order, which shall be followed by the pixels in the remaining rows in increasing row order. The pixels in each row appear in similar fashion to those of the first row. The bit-mapped symbol data shall end with the LSB of the last pixel of the last line. No field delimiters shall be used to designate the end of the symbol data. The last bit of a symbol shall be zero padded to the next byte boundary and followed by the first character of the next subheader.

5.6.2.2 Object symbols. The use of object graphics substantially reduces the amount of data that must be stored to display symbols properly. Objects are predefined symbols contained in a library with standardization contents. The entries in the library may be indexed by number, as shown in table X. In future versions of the NITF, object symbols will be referenced by an alphanumeric code. The determination of the methodology to be used to draw the object is implementation specific (that is, the NITF specifies what is to be drawn and its parameters, while the decision as to how to draw it is left to the implementor). The number of objects currently defined is quite limited, although the list will be expanded substantially in the future. All the data required to specify or describe an object symbol are stored in the symbol subheader. Thus, no data field follows the subheader for an object symbol.

TABLE X. Object symbol descriptions.

TYPE	DESCRIPTION	ROTATION POINT
1	Solid Line color=SCOLOR, width=NWDTH From SLOC, to SLOC2	None
2	Dashed Line Alternating 10 Pixels SCOLOR, 10 Pixels Transparent From SLOC, to SLOC2, width=NWDTH	None
3	Dashed Line Alternating 15 Pixel SCOLOR, 10 Pixels Transparent From SLOC, to SLOC2, width=NWDTH	None
4	Dashed Line Alternating 20 Pixels SCOLOR, 5 Pixels Transparent From SLOC, to SLOC2, width=NWDTH	None
5	Dashed Line Alternating 5 Pixels SCOLOR, 5 Pixels Transparent From SLOC, to SLOC2, width=NWDTH	None
6	Dashed Line Alternating 10 Pixels SCOLOR, 5 Pixels Transparent, 5 Pixels SCOLOR, 5 Pixels Transparent, From SLOC, to SLCO2, width=NWDTH	None
7	Reserved	
8	Reserved	
9	Reserved	
10	Triangle An isosceles triangle of width NPIXPL, height NLIPS, edge width NWDTH, color=SCOLOR, rotated about SLOC to angle SROT, Apex=SLOC, SLOC2=0000000000	Apex
11	Rectangle A rectangle of width NPIXPL, height NLIPS, edge width NWDTH, color=SCOLOR, rotated about SLOC to angle SROT, center=SLOC, SLOC2=0000000000	Center

TABLE X. Object symbol descriptions - Continued.

TYPE	DESCRIPTION	ROTATION POINT
12	Square A square, width NPIXPL=height NLIPS, edge width=NWDTH, color=SCOLOR, rotated about SLOC to angle SROT, center=SLOC, SLOC2=0000000000	Center
13	Circle A circle, diameter=NPIXPL, NLIPS=NPIXPL, edge width=NWDTH, color=SCOLOR, center=SLOC, SLOC2=0000000000	None
14	Ellipse An ellipse, semimajor axis=NPIXPL, semiminor axis=NLIPS, center=SLOC, edge width=NWDTH, color=SCOLOR, semimajor axis rotated about center to SROT, SLOC2=0000000000	Center
15	Bracket A vertical square bracket, nibs pointing right, height=NLIPS, Nib Width=NPIXPL, center of shaft at SLOC, edge width=NWDTH, color=SCOLOR, rotated about SLOC to angle SROT, SLOC2=0000000000	Center of Shaft
16	Diamond A vertical diamond, height=NLIPS, width=NPIXPL, center=SLOC, edge width=NWDTH, color=SCOLOR, SLOC2=0000000000	Center
17	Hexagon An equilateral hexagon, center = SLOC, width = NPIXPL, height = NLIPS, edge width = NWDTH, color = SCOLOR, SLOC2 = 0000000000	Center
18	Arrow An arrow, with head elements length of NPIXPL at 45^{o} angle to shaft, shaft width = NWDTH, color = SCOLOR, base of shaft = SLOC, apex of arrowhead = SLOC2	Bottom of Vertical Shaft

5.6.2.3 CGM symbols. A CGM symbol graphic is a CGM "file" preceded by the symbol-subheader. The precise tailoring of the CGM standard to NITF is found in MIL-STD-2301. General information on CGM can be found in FIPS PUB 128, Computer Graphics Metafile (CGM) [adaptation of American National Standards Institute (ANSI) X3.122-1986].

5.7 Label data type. The label data type in the NITF is used to store a label composed of printable ASCII characters plus carriage returns and line feeds. The intent is for the label to be nondestructively overlaid upon one or more images and/or symbols to serve as textual annotation. However, it is recommended that label information be presented using the CGM symbol data type construct rather than the label data type construct. The label data type may not be included in future releases of the standard.

5.7.1 Label subheader. The label subheader is used to identify and supply information about the label necessary to read and display the label. The format for the label subheader is shown in table XI. Descriptions of the subheader parameters follow in table XII.

TABLE XI. Label subheader.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	FORMAT VALUE	TYPE
LA	File Part Type	2	LA	R
LID	Label ID	10	Alphanumeric	R
LSCLAS	Label Security Classification	1	T, S, C, R, or U	R
LSCODE	Label Codewords	40	Alphanumeric	O
LSCTLH	Label Control and Handling	40	Alphanumeric	O
LSREL	Label Releasing Instructions	40	Alphanumeric	O
LSCAUT	Label Classification Authority	20	Alphanumeric	O
LSCTLN	Label Security Control Number	20	Alphanumeric	O
LSDWNG	Label Security Downgrade	6	Alphanumeric	O
LSDEVT	Label Downgrading Event	40	Alphanumeric	C
ENCRYP	Encryption	1	0=Not Encrypted 1=Encrypted	R
LFS	Label Font Style	1	Alphanumeric	R
LCW	Label Cell Width	2	1-99	O
LCH	Label Cell Height	2	1-99	O
LDLVL	Display Level	3	1-999	R
LALVL	Attachment Level	3	0-998	R
LLOC	Label Location	10	rrrrrccccc	R

TABLE XI. Label subheader - Continued.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	FORMAT VALUE	TYPE
LTC	Label Text Color	3	1 Byte of R, G, and B	R
LBC	Label Background Color	3	1 Byte of R, G, and B	R
LXSHDL	Extended Subheader Data Length	5	0-09747	R
LXSOFL	Extended Subheader Overflow	3	0-999	C
LXSHD	Extended Subheader Data	*	Controlled Extension Tagged Records	

* As specified in LXSHDL

TABLE XII. NITF label subheader fields.

LA	This field shall contain the characters "LA" to identify the subheader as a label subheader.
LID	This field shall contain a valid alphanumeric identification code associated with the label. The valid codes are determined by the application.
LSCLAS	This field shall contain a valid value representing the classification level of the image. Valid values are: T (=Top Secret), S (=Secret), C (=Confidential), R (=Restricted), U (=Unclassified).
LSCODE	This field shall contain a valid indicator of the security compartments associated with the label. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, and complete words and project numbers. The selection of a relevant set of codewords and project numbers is application specific. If this field is all spaces, it shall imply that no codewords apply to the label.
LSCTLH	This field shall contain valid security handling instructions associated with the label. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, complete words and abbreviations of more than two characters, and phrases only if the words within the phrase are separated by hyphens. The selection of a relevant set of security handling instructions is implementation specific. If this field is all spaces, it shall imply that no label control and handling instructions apply.

TABLE XII. NITF label subheader fields - Continued.

LSREL	This field shall contain a valid list of countries and/or groups of countries to which the label is authorized for release. Valid items in the list are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: country codes and groupings that are digraphs in accordance with FIPS PUB 10-3. If this field is all spaces, it shall imply that no label release instructions apply.
LSCAUT	This field shall contain a valid identity code of the classification authority for the label. The code shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no label classification authority applies.
LSCTLN	This field shall contain a valid security control number associated with the label. The format of the security control number shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no label security control number applies.
LSDWNG	This field shall contain a valid indicator that designates the time at which a declassification or downgrading action is to take place (in accordance with table V) Valid values are (1) the calendar date in the format YYMMDD, (2) the code "999999" when the originating agency's determination is required (OADR), and (3) the code "999998" when a specific event determines at what time declassification or downgrading is to take place. If this field is all spaces, it shall imply that no label security downgrade condition applies.
LSDEVT	If the Label Security Downgrade field (LSDWNG) equals "999998," this field shall be present and shall contain a valid specification of the downgrading event. If this field is present and all spaces, it shall imply that an error exists. Valid values for the event specification are determined by the application.
ENCRYP	This field shall contain the value zero until such time as this specification is updated to defined the use of other values.
LFS	This field is reserved for future use. It shall contain one space (ASCII 32, decimal) by default.
LCW	This field shall contain the width in pixels of the character cell (rectangular array used to contain a single character in monospaced fonts) used by the file originator. The default value of 00 indicates the file originator has not included this information.
LCH	This field shall contain the height in pixels of the character cell (rectangular array used to contain a single character in monospaced fonts) used by the file originator. The default value of 00 indicates the file originator has not included this information.

TABLE XII. NITF label subheader fields - Continued.

LDLVL	This field shall contain a valid value that indicates the graphic display level of the label relative to other displayed file components in a composite display. The valid values are 001 to 999. The display level of each displayable file component (image, label, symbol) within a file shall be unique; that is, each number from 001 to 999 is the display level of, at most, one item. The meaning of display level is discussed fully in 5.3.3. The symbol, image, or label component in the file having the minimum display level shall have attachment level zero (ILOC, SLOC, and LLOC field descriptions).
LALVL	This field shall contain a valid value that indicates the attachment level of the label. Valid values for this field are 0 and the display level value of any other image, symbol, or label in the file. The meaning of attachment value is discussed fully in 5.3.4. The symbol, image, or label component in the file having the minimum display level shall have attachment level zero (ILOC, SLOC, and LLOC field descriptions).
LLOC	A label's location specified by providing the location of the upper left corner of the minimum bounding rectangle of the label. This field shall contain the label location represented as rrrrrccccc, where rrrrr and ccccc are the row and the column offset from the ILOC, SLOC, or LLOC value of the item to which the label is attached. A row or column value of 00000 indicates no offset. Positive row and column values indicate offsets down and to the right and range from 00001 to 99999, while negative row and column values indicate offsets up and to the left and must be within the range -0001 to -9999. The coordinate system used to express ILOC, SLOC, and LLOC fields shall be common for all images, labels, and symbols in the file having attachment level zero. The location in this common coordinate system of all displayable graphic components can be computed from the offsets given in the ILOC, SLOC, and LLOC fields.
LTC	The interpretation of RGB color shall be the same as defined in Table VIII. This field shall contain three bytes to be interpreted as the red, green, and blue components (respectively the first, second, and third bytes) of the color of the text characters in the label. The byte values shall be interpreted as integers in the range 0 to 255.
LBC	The interpretation of RGB color shall be the same as defined in Table VIII. This field shall contain three bytes to be interpreted as the red, green, and blue components (respectively the first, second, and third bytes) of the label background color. The label background comprises those pixels within the minimum bounding rectangle of the label text that are not part of the text characters. The byte values shall be interpreted as integers in the range 0 to 255.

TABLE XII. NITF label subheader fields - Continued.

LXSHDL	This field shall contain the length of bytes of the sum of the following two fields (LXSOFL + LXSHD). This length is three plus the sum of the lengths of all the controlled tagged record extensions (see 5.9) appearing in the LXSHD field. A value of zero shall mean that no controlled tagged record extensions are included in the label subheader. If a controlled tagged record extension is too long to fit in the LXSHD field, it shall be put in a data extension segment (see 5.9).
LXSOFL	If present, this field shall contain "000" if the tagged record extensions in LXSHD do not overflow into a DES, or shall contain the sequence number in the file of the DES into which they do overflow. This field shall be omitted if the field LXSHDL contains zero.
LXSHD	If present, this field shall contain controlled tagged record extension (see 5.9) approved and under configuration management by the NTB. The length of this field shall be the length specified by the field LXSHDL, less the length (3) of LXSOFL. Controlled tagged record extensions in this field for a label shall contain information pertaining specifically to the label. Controlled tagged record extensions shall appear one after the other in this field with no intervening bytes. The first byte of this field shall be the first byte of the first controlled tagged record extension appearing in the field. The last byte of this field shall be the last byte of the last controlled tagged record extension to appear in the field. This field shall be omitted if the field LXSHDL contains zero.

5.7.2 Label data. Labels are to be used to display ASCII characters overlaid on an image in a system's native text display mode. Requirements for expressing a label in a precise font and style and with elaborate backgrounds may be met by defining the label as a bit-mapped symbol. The data contained in each label in the file shall follow the corresponding label subheader without intervening bytes. The label shall be presented in the file as contiguous data with each ASCII character immediately following the other. The label data shall begin with the first, or the left-most character of the label text, followed by subsequent characters as read from left to right. For multiple line labels, a carriage return followed by a line feed shall be used to delimit lines in the label, where the first character of the next line shall follow the ASCII line feed character immediately and, when displayed, shall be placed immediately below the first character of the preceding line. The label data shall end with the last character of the label. No field delimiters or special characters shall be used to designate the end of the label data. If more than one label is in the file, the last character of the first label shall be followed by the first character of the second label subheader. Care should be taken to ensure that all label information fits within the limits of the images and symbols in the file to be displayed. This can be done by using accurate label size information in the appropriate label subheader fields.

5.8 Text data type. The text data field shall be used to store a file or item of text, such as a word processing file or document. Text items are intended to convey information about the image product contained in the NITF file.

5.8.1 Text data subheader. The text subheader is used to identify and supply the information about the text file necessary to read and display the text. The format for the text subheader is shown in table XIII. Descriptions of the subheader parameters follow in table XIV.

TABLE XIII. Text subheader.

(R) = required, (O) = optional, and (C) = conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
TE	File Part Type	2	TE	R
TEXTID	Text ID	10	Alphanumeric	R
TXTDT	Text Date & Time	14	DDHHMMSSZMONYY	R
TXTITL	Text Title	80	Alphanumeric	O
TSCLAS	Text Security Classification	1	T, S, C, R, or U	R
TSCODE	Text Codewords	40	Alphanumeric	O
TSCTLH	Text Control and Handling	40	Alphanumeric	O
TSREL	Text Releasing Instructions	40	Alphanumeric	O
TSCAUT	Text Classification Authority	20	Alphanumeric	O
TSCTLN	Text Security Control Number	20	Alphanumeric	O
TSDWNG	Text Security Downgrade	6	Alphanumeric	O
TSDEVT	Text Downgrading Event	40	Alphanumeric	O
ENCRYP	Encryption	1	0=Not Encrypted 1=Encrypted	R
TXTFMT	Text Format	3	JTC, STA, OTH	R
TXSHDL	Extended Subheader Data Length	5	0-09677	R
TXSOFL	Extended Subheader Overflow	3	0-999	C
TXSHD	Extended Subheader Data	*	Alphanumeric	C

*As specified by the value in the TXSHDL field

TABLE XIV. NITF text subheader fields.

TE	This field shall contain the characters "TE" to identify the subheader as a text subheader.
TEXTID	This field shall contain a valid alphanumeric identification code associated with the text item. The valid codes are determined by the application.
TXTDT	This field shall contain the time (Zulu) of origination of the text in the format DDHHMMSSZMONYY where DD is the day of the month (01-31), HH is the hour, (00-23), MM is the minute (00-59), SS is the second (00-59), the character Z is required, MON is the first three characters of the month, and YY is the last two digits of the year.
TXTITL	This field shall contain the title of the text item.
TSCLAS	This field shall contain a valid value representing the classification level of the text item. Valid values are: T (=Top Secret), S (=Secret), C (=Confidential), R (=Restricted), U (=Unclassified).
TSCODE	This field shall contain a valid indicator of the security compartments associated with the text item. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, and complete codewords or project numbers. The selection of a relevant set of codewords and project numbers is application specific. If this field is all spaces, it shall imply that no codewords apply to the text item.
TSCTLH	This field shall contain valid security handling instructions associated with the text item. Valid values are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: digraphs in accordance with table V, trigraphs not contained in table V, complete words and abbreviations of more than two characters, and phrases only if the words within the phrase are separated by hyphens. The selection of a relevant set of security handling instructions is implementation specific. If this field is all spaces, it shall imply that no text control and handling instructions apply.
TSREL	This field shall contain a valid list of countries and/or groups of countries to which the text item is authorized for release. Valid items in the list are one or more of the following separated by single spaces (ASCII 32, decimal) within the field: country codes and groupings that are digraphs in accordance with FIPS PUB 10-3. If this field is all spaces, it shall imply that no text release instructions imply.
TSCAUT	This field shall contain a valid identity code of the classification authority for the text item. The code shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no text classification authority applies.

TABLE XIV. NITF text subheader fields - Continued.

TSCTLN	This field shall contain a valid security control number associated with the text item. The format of the security control number shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all spaces, it shall imply that no text security control number applies.
TSDWNG	This field shall contain a valid indicator that designates the time at which a declassification or downgrading action is to take place, in accordance with table V. Valid values are (1) the calendar date in the format YYMMDD, (2) the code "999999" when the originating agency's determination is required (OADR), and (3) the code "999998" when a specific event determines at what time declassification or downgrading is to take place. If this field is all spaces, it shall imply that no text security downgrade condition applies.
TSDEVT	If the Text Security Downgrade field (TSDWNG) equals "999998," then this field shall be present and shall contain a valid specification of the downgrade event. If this field is present and all spaces, it shall imply that an error exists. Valid values for the event specification are determined by the application.
ENCRYP	This field shall contain the value zero until such time as this specification is updated to define the use of other values.
TXTFMT	This field shall contain a valid three-character code indicating the format or template to be used to display the text. Valid codes are MTF to indicate USMTF (Refer to JCS PUB 6-04 for examples of the USMTF format), STA to indicate NITF ASCII, and OTH to indicate other, such as user-defined. Refer to appendix A for additional discussion of standard and NITF ASCII.
TXSHDL	This field shall contain the length in bytes of the sum of the following two fields (TXSOFL + TXSHD). This length is three plus the sum of the lengths of all the controlled tagged record extensions (see 5.9) appearing in the TXSHD field. A value of zero shall mean that no controlled tagged record extensions are included in the text subheader. If a controlled tagged record extension is too long to fit in the TXSHD field, it shall be put in a data extension segment (see 5.9).
TXSOFL	If present, this field shall contain "000" if the tagged record extensions in TXSHD do not overflow into a DES, or shall contain the sequence number in the file of the DES into which they do overflow. This field shall be omitted if the field TXSHDL contains zero.

TABLE XIV. NITF text subheader fields - Continued.

TXSHD	If present, this field shall contain controlled tagged record extensions (see 5.9) approved and under configuration management by the NTB. The length of this field shall be the length specified by the field TXSHDL, less the length (3) of TXSOFL. Controlled tagged record extensions in this field shall contain information pertaining specifically to the text. Controlled tagged record extensions shall appear one after the other in this field with no intervening bytes. The first byte of this field shall be the first byte of the first controlled tagged record extension appearing in the field. The last byte of this field shall be the last byte of the last controlled tagged record extension to appear in the field. This field shall be omitted if the field TXSHDL contains zero.

5.8.2 Text data. Text is used to display ASCII characters in a system's native mode. The data field containing a text item included in an NITF file shall follow the corresponding text subheader. The text data shall consist entirely of characters permitted by the text format specified in the subheader.

5.8.2.1 NITF ASCII data representation. The NITF ASCII format is composed of the following ASCII characters (all numbers are decimal: Line Feed (10), Form Feed (12), Carriage Return (13), and space (32) through tilde (126)). This set includes all of the alphanumeric characters as well as all commonly used punctuation characters. All lines within an NITF ASCII file shall be separated by carriage return/line feed pairs. For NITF ASCII, the text data shall be presented as a contiguous file with each permitted ASCII character immediately following the other. The text data shall begin with the first or left-most character of the text, followed by subsequent characters as read from left to right. A carriage return followed by a line feed shall be used to delimit lines in the text where the first character from the next line immediately follows the ASCII line feed character. The text data shall end with the last character of the text. No field delimiters or special characters shall be used to designate the end of the text data file. If more than one text item is included in an NITF file, the last character of the first text item shall be followed by the first character of the second text subheader.

5.9 Future expansion. Future expansion of the NITF is supported in two ways: (1) built-in mechanisms and procedures to allow immediate inclusion of user-determined and user-defined data characteristics and kinds of data without changing this standard, and (2) a collection of data fields called Reserved Extension Segments providing space within the file structure for entirely unspecified future purposes. Addition of further data characteristics beyond those specified in this standard is accomplished using the User Data (UDHD and UDID), Extended Header Data (XHD), and Extended Subheader Data (IXSHD, SXSHD, LXSHD, and TXSHD) fields. Use of these fields is described in 5.9.1.1 and 5.9.1.2. Addition of new kinds of data items is accomplished using Data Extension Segments defined in 5.9.1.3.1. Extensions of all types may be incorporated into the file while maintaining backward compatibility, since the byte count mechanisms provided allow applications developed prior to the addition of newly defined data, or to simply skip over extension fields they are not designed to interpret.

5.9.1 Tagged record extensions. Variations of the same basic extension mechanism, tagged records, are used for all extensions except the Reserved Extension Segments, which will be discussed separately. There are three varieties of tagged record extensions: registered extensions, controlled extensions, and encapsulated extensions. Figure 9 illustrates the concepts and formatting descriptions in 5.9.1.1 through 5.9.1.3. A current listing of the tagged record extensions that have been registered with the NTB is provided in Appendix B.

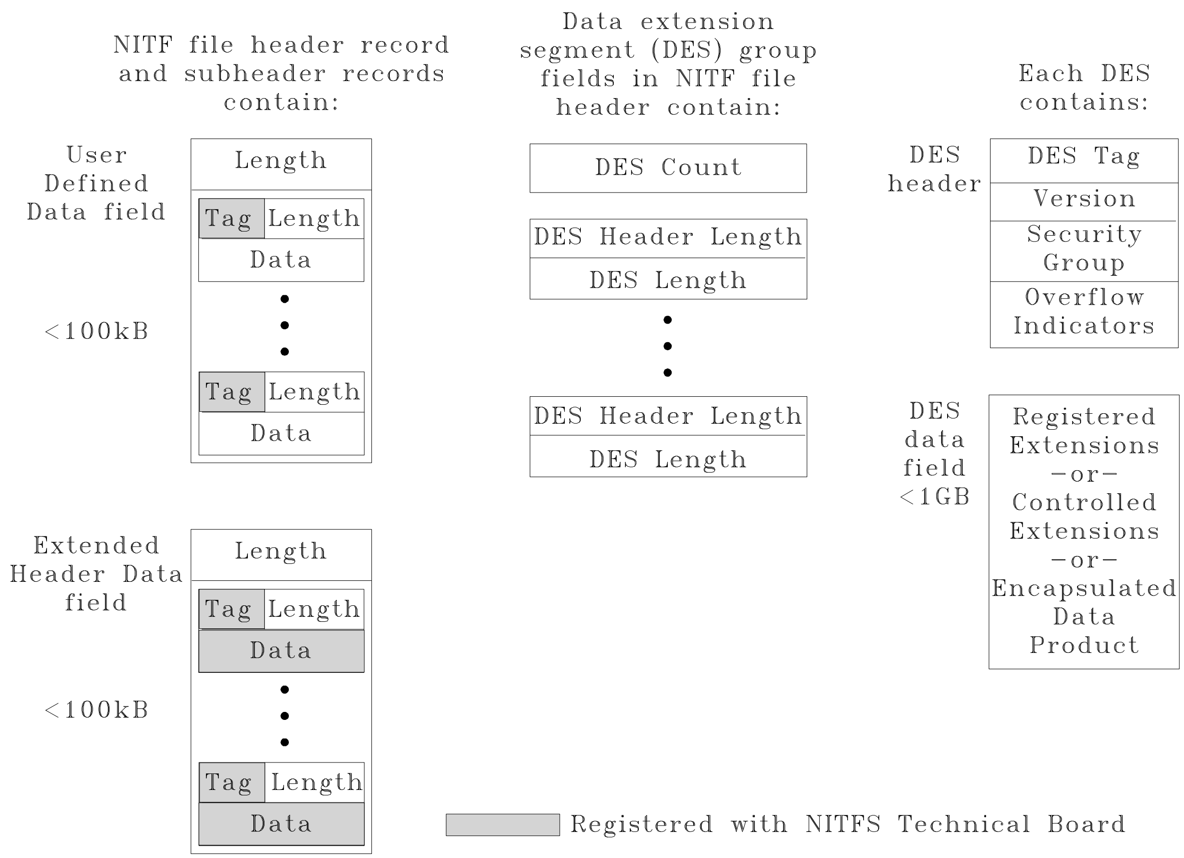


FIGURE 9. Tagged record and data extension segment formats.

5.9.1.1 Registered extensions. Each registered tagged record extension consists of three required fields. These fields are defined in tables XV and XVI. These extensions are user-defined, and only the six character RETAG field is registered with the NTB. The purpose of registering the tags is to avoid having two users use the same tag to mean different extensions. A sequence of registered tagged record extensions can appear in the NITF header User Defined Data field, UDHD, or any image subheader in its User Defined Image Data field, UDID. When the tagged record extension carries data associated with the file as a whole, it should appear in the UDHD field, if sufficient room is available. If the extension carries data associated with an image data item in the file, it should appear in the UDID field of that item's subheader, if sufficient room is available. A registered tagged record extension may appear in a Data Extension Segment (see 5.9.1.3 and subparagraphs) that is designated to contain registered tagged record extensions, but only if sufficient space is not available in the UDHD or a UDID, as appropriate. A registered tagged record extension shall be included in its entirety within the UDHD, a single UDID or the single DES selected to contain it. A registered tagged record extension may not "overflow" file fields.

TABLE XV. Registered tagged record extension format.

(R) = required, (O) = optional, and (C) conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
RETAG	Unique extension type identifier	6	Alphanumeric	R
REL	Length of REDATA field	5	1 to 99988	R
REDATA	User-defined data	*	User-defined	R

*As indicated in REL field

TABLE XVI. Registered tagged record extension field descriptions.

FIELD	VALUE DEFINITIONS AND CONSTRAINTS
RETAG	This field shall contain a valid alphanumeric identifier properly registered with the NTB.
REL	This field shall contain the length in bytes of the data contained in REDATA. The Tagged record's length is 11 + REL.
REDATA	This field shall contain data of either binary or character data types defined by and formatted according to user specification. The length of this field shall not cause any other NITF field length limits to be exceeded, but is otherwise fully user-defined.

5.9.1.2 Controlled extensions. These extensions are defined and submitted to the NTB for approval by the NTB and, once accepted are subject to configuration management by the NTB. They are documented in a series of documents maintained by the NTB. The tagged record format for controlled extensions is identical to that for registered extensions (detailed in table XV and table XVI) except that the first two letters of each field identifier change from "RE" to "CE." The six character CETAG field and the structure of the CEDATA data field shall be registered and configuration controlled. A sequence of controlled tagged record extensions can appear in the XHD field of the NITF file header or in the IXSHD, SHSHD, LXSHD, or TXSHD field of a standard data type data item in the file. When the controlled tagged record extension carries data that is associated with the file as a whole, it should appear in the XHD field, if sufficient room is available. If the extension carries data associated with a data item in the file, it should appear in the IXSHD, SHSHD, LXSHD, or TXSHD field of that item's subheader, if sufficient room is available. A controlled tagged record extension may appear in a Data Extension Segment (see 5.9.1.3 and subparagraphs), which is designated to contain controlled tagged record extensions, but only if appropriate. A controlled tagged record extension shall be included in its entirety within the XHD, a single IXSHD, SHSHD, LXSHD, or TXSHD or the single DES selected to contain it. A single controlled tagged record extension may not "overflow" file fields.

5.9.1.3 Encapsulated extensions. These extensions are similar to the registered extensions in that each has a tag, and in this case, the tag versions are registered with the NTB. Each encapsulated extension shall appear in its own Data Extension Segment (DES) and shall conform to the DES structure (see 5.9.1.3.1). There are two reserved tags: "Registered Extensions" and "Controlled Extensions." These tags are for use when a series of registered or controlled, tagged record extensions is to appear in a DES (see 5.9.1.1 and 5.9.1.2) as "overflow" from the NITF file header or any subheader. Which header or subheader overflowed is indicated in the DESOFLOW and DESITEM field contents. Generally, the data in an encapsulated extension is user-defined. The data are anticipated to be defined typically by a specific version of a specific standard or product specification (which may or may not be under the control of the NTB). Encapsulated extensions allow the incorporation of data products in an NITF file to be disseminated along with an image. For example, Digital Terrain Elevation Data (DTED), Digital Feature Analysis Data (DFAD), or other DMA products could be distributed along with an image product to support analysis and interpretation of the image. Audio and video segments are additional examples of data that may be added to the NITF through the use of Data Extension Segments.

5.9.1.3.1 Data extension segment structure. The NITF header accommodates up to 999 DES. Each DES shall consist of a DES subheader and a DES data field (similar to the way a standard data type data item has a data field and an adjacent associated subheader). Within the Data Extension Segment Group in the NITF Header Record is found the number of DES in the file, the length of each DES subheader, and length of the DES data field, DESDATA. The field size specifications in the NITF file header allow each DES to be just less than one gigabyte in length. The DES subheader shall contain the fields defined in table XVII and table XVIII. The structure provided in the DES by the fields DESSHL, DESSHF, and DESDATA is intended to encourage the formation of DES along the lines of the standard data types in the NITF, in which a group of ASCII fields describing the data is followed by the data itself.

TABLE XVII. Data extension segment subheader format.

(R) = required, (O) = optional, and (C) conditional

FIELD	NAME	SIZE	VALUE RANGE	TYPE
DE	File Part Type	2	DE	R
DESTAG	Unique DES type identifier	25	Alphanumeric	R
DESVER	Version of the data field definition	2	1 to 99	R
DESSG	Security group	†	(See Table XVIII)	R
DESOFLW	Overflowed header type	6	Alphanumeric	C
DESITEM	Data item overflowed	3	0 to 999	C
DESSHL	Length of user-defined subheader fields	4	0-9999	R
DESSHF	User-defined subheader fields	*	Alphanumeric	C
DESDATA	User-defined data field	**	User defined	R

† 167 or 207 - table XVIII for explanation
* Value specified in DESSHL
** Determined by user. If DESTAG = "Registered Extensions" or "Controlled Extensions," this signifies the sum of the lengths of the included tagged records.

TABLE XVIII. Data extension segment subheader field definitions.

FIELD	VALUE DEFINITIONS AND CONSTRAINTS
DE	This field shall contain the characters "DE" to identify the subheader as a data extension.
DESTAG	This field shall contain a valid alphanumeric identifier properly registered with the NTB.
DESVER	This field shall contain the alphanumeric version number of the use of the tag. The NTB assigns the version number as part of the registration process.
DESSG	This field shall contain a series of fields containing security classification information for the DES as a whole. The fields included shall mirror those of the NITF file header from FSCLAS through FSDEVT, including field length and content, but be applicable to the DES only. The field names shall be DESCLAS through DESDEVT respectively, simply substituting "DE" for "F." The number of bytes consumed by this field group will be 167 or 207, depending on whether the conditional DESDEVT field is present.

TABLE XVIII. Data extension segment subheader field definitions - Continued.

DESOFLW	This field shall be present if DESTAG = "Registered Extensions" or "Controlled Extensions." Its presence indicates that the DES contains a tagged record extension that would not fit in the file header or component header where it would ordinarily be located. Its value indicates the data type to which the enclosed tagged record is relevant. If the value of DESTAG is "Controlled Extensions," the valid values for DESOFLOW are XHD, IXSHD, SXSHD, LSXHD or TXSHD. If the value of DESTAG is "Registered Extensions," the valid values for DESOFLW are UDHD and UDID.
DESITEM	This field shall be present if DESOFLW is present. It shall contain the number of the data item in the file, of the type indicated in DESOFLW to which the tagged record extensions in the segment apply. For example, if DESOFLW = UDID and DESITEM = 3, then the tagged record extensions in the segment applies to the third image in the file. If the value of DESOFLW = UDHD, the value of DESITEM shall be 0.
DESSHL	This field shall contain the number of bytes in the field DESSHF. If this field contains 0, DESSHF shall not appear in the DES subheader. This field shall contain 0 if DESTAG = "Registered Extensions" or "Controlled Extensions".
DESSHF	This field shall contain user-defined fields. Data in this field shall be alphanumeric, formatted according to user specification.
DESDATA	This field shall contain data of either binary or character types defined by and formatted according to the user's specification. However, if the DESTAG is "Registered Extensions" or "Controlled Extensions," the tagged records shall appear according to their definition with no intervening bytes. The length of this field shall not cause any other NITF field length limits to be exceeded, but is otherwise fully user-defined.

5.9.2 Reserved extension segments. Structure is provided in the NITF file header to support up to 999 distinct fields of up to 9999999 bytes plus a corresponding subheader of up to 9999 bytes for each field. The combination of each subheader and corresponding data field is called a Reserved Extension Segment. These fields are reserved in that they shall not be present in any header until this standard is modified to define their use. See the definition of the field NUMRES and following field in tables II and III.

6. NOTES

(This section contains general or explanatory information that may be helpful but is not mandatory.)

6.1 Use of NITF in NITFS. Though the NITF was conceived initially to support the transmission of a message composed of a primary image, image insets or subimages, symbols, labels, and text, its current form makes it suitable for a wide variety of file exchange needs. One of the flexible features of the NITF is that it allows several items of each data type to be included in one file, yet any data types may be omitted. Thus, for example, the NITF may equally well be used for the storage of a single portion of text, a single image or a complex composition of several images, symbols, labels and text. The following section discusses an example NITF file of moderate complexity.

6.2 Example file. Table XIX shows the contents of the fields in the header of a sample NITF file composed of one base image, one inset image overlay, five symbols, two labels, and five text selections. Figure 10 shows the sample file as a composite image with its overlay graphics and labels. In an NITF file, the data for each data item is preceded by the item's subheader. The subheader for a data type is omitted if no items of that type are included in the file. Sample subheaders for items in the sample file are shown in Table XX through Table XXIX.

TABLE XIX. Example NITF file header.

NITF HEADER FIELD	FORMAT	COMMENT
File Type & Version (FHDR)	NITF02.00	9 characters
Compliance Level (CLEVEL)	04	2 characters -- images less than or equal to 4k x 4k
System Type (STYPE)		4 blank characters
Originating Station ID (OSTAID)	U21SOO90	8 characters with 2 spaces
File Date & Time (FDT)	03191639ZMAR93	14 characters
File Title (FTITLE)	This NITF file contains 2 images, 5 symbols, 2 labels, and 5 text files.	72 characters followed by 8 spaces - 80 characters
File Security Classification (FSCLAS)	U	1 character
File Codewords (FSCODE)		40 spaces
File Control & Handling (FSCTLH)		40 spaces
File Releasing Instructions (FSREL)		40 spaces

TABLE XIX. Example NITF file header - Continued.

NITF HEADER FIELD	FORMAT	COMMENT
File Classification Authority (FSCAUT)		20 spaces
File Security Control Number (FSCTLN)		20 spaces
File Security Downgrade (FSDWNG)	999998	6 characters
File Downgrade Event (FSDEVT)	This file will not need a downgrade.	36 characters followed by 4 spaces
File Copy Number (FSCOP)	00001	5 digits
File Number of Copies (FSCPYS)	00001	5 digits
Encryption (ENCRYP)	0	Required default
Originator's Name (ONAME)	NITF Technical Board	20 characters followed by 7 spaces - 27 characters
Originator's Phone Number (OPHONE)	(555)555-5555	13 characters followed by 5 spaces - 18 characters
File Length (FL)	000001052271	12 digits
NITFS File Header Length (HL)	000583	6 digits
Number of Images (NUMI)	002	3 digits
Length of 1st Image Subheader (LISH001)	000723	6 digits
Length of 1st Image (LI001)	0000503889	10 digits (compression factor of 12)
Length of 2nd Image Subheader (LISH002)	000439	6 digits
Length of 2nd Image (LI002)	0000430336	10 digits
Number of Symbols (NUMS)	002	3 digits
Length of 1st Symbol Subheader (LSSH001)	0298	4 digits
Length of 1st Symbol (LS001)	001046	6 digits
Length of 2nd Symbol Subheader (LSSH002)	0298	4 digits

TABLE XIX. Example NITF file header - Continued.

NITF HEADER FIELD	FORMAT	COMMENT
Length of 2nd Symbol (LS002)	000902	6 digits
Length of 3rd Symbol Subheader (LSSH003)	0298	4 digits
Length of 3rd Symbol (LS003)	008647	6 digits
Length of 4th Symbol Subheader (LSSH004)	0298	4 digits
Length of 4th Symbol (LS004)	001010	6 digits
Length of 5th Symbol Subheader (LSSH005)	0298	4 digits
Length of 5th Symbol (LS005)	001068	6 digits
Number of Labels (NUML)	002	3 digits
Length of 1st Label Subheader (LLSH001)	0252	4 digits
Length of 1st Label (LL001)	013	3 digits
Length of 2nd Label Subheader (LLSH002)	0252	4 digits
Length of 2nd Label (LL002)	011	3 digits
Number of Text Files (NUMT)	005	3 digits
Length of 1st Text Subheader (LTSH001)	0322	4 digits
Length of 1st Text File (LT001)	20000	5 digits
Length of 2nd Text Subheader (LTSH002)	0322	4 digits
Length of 2nd Text File (LT002)	20000	5 digits
Length of 3rd Text Subheader (LTSH003)	0322	4 digits
Length of 3rd Text File (LT003)	20000	5 digits
Length of 4th Text Subheader (LTSH004)	0322	4 digits
Length of 4th Text File (LT004)	20000	5 digits

TABLE XIX. Example NITF file header - Continued.

NITF HEADER FIELD	FORMAT	COMMENT
Length of 5th Text Subheader (LTSH005)	0322	4 digits
Length of 5th Text File (LT005)	20000	5 digits
Number of Data Extension Segments (NUMDES)	000	3 digits
Number of Reserved Data Extension Segments (NUMRES)	000	3 digits
User Defined Header Data Length (UDHDL)	00000	5 digits
Extended Header Data Header Length (XHDL)	00000	5 digits

TABLE XX. Example image subheader of the base image.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (IM)	IM	2 characters
Image ID (IID)	0000000001	10 characters
Image Date & Time (IDATIM)	01120000ZJAN89	14 characters
Target ID (TGTID)		17 spaces
Image Title (ITITLE)	This is an unclassified base image in an unclassified NITF file.	64 characters followed by 16 spaces - 80 characters total
Image Security Classification (ISCLAS)	U	1 character
Image Codewords (ISCODE)		40 spaces
Image Control and Handling (ISCTLH)		40 spaces
Image Releasing Instructions (ISREL)		40 spaces
Image Classification Authority (ISCAUT)		20 spaces

TABLE XX. Example image subheader of the base image - Continued.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
Image Security Control Number (ISCTLN)		20 spaces
Image Security Downgrade (ISDWNG)	999998	6 characters
Image Downgrading Event (ISDEVT)	This image will not need downgrading.	37 characters followed by 3 spaces - 40 total characters
Encryption (ENCRYP)	0	Required default
Image Source (ISORCE)		42 spaces
Number of Significant Rows in image (NROWS)	00002756	8 characters
Number of Significant Columns in image (NCOLS)	00002194	8 characters
Pixel Value Type (PVTYPE)	INT	3 characters - interpret pixel values as integers
Image Representation (IREP)	MONO	4 characters followed by 4 spaces - grayscale imagery
Image Class (ICAT)	VIS	3 characters followed by 5 spaces - visible imagery
Actual Bits-Per-Pixel Per Band (ABPP)	08	2 digits
Pixel Justification (PJUST)	R	1 character
Image Coordinate System (ICORDS)	N	1 character - indicates no geo location coordinates
Number of Image Comments (NICOM)	3	1 digit
Image Comment 1 (ICOM1)	This is the first comment.	25 characters followed by 55 spaces - 80 total characters
Image Comment 2 (ICOM2)	This is the second comment.	26 characters followed by 54 spaces - 80 total characters
Image Comment 3 (ICOM3)	This is the third comment.	25 characters followed by 55 spaces - 80 total characters

TABLE XX. Example image subheader of the base image - Continued.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
Image Compression (IC)	C3	2 characters - indicates JPEG compressed
Compression Rate Code (COMRAT)	00.1	4 characters - indicates general, quality level 1
Number of Bands (NBANDS)	1	1 digit
1st Band Representation (IREPBAND1)		2 spaces
1st Band Significance for Image Category (ISUBCAT1)		6 spaces
1st Band Image Filter Condition (IFC1)	N	1 character - required default value
1st Band Standard Image Filter Code (IMFLT1)		3 spaces - reserved
1st Band Number of LUTS (NLUTS1)	0	1 character
Image Sync Code (ISYNC)	0	1 digit
Image Mode (IMODE)	B	1 character - B required for 1 band
Number of Blocks per Row (NBPR)	0001	4 digits
Number of Blocks per Column (NBPC)	0001	4 digits
Number of pixels Per Block Horizontal (NPPBH)	2194	4 digits
Number of pixels Per Block Vertical (NPPBV)	2756	4 digits
Number of Bits per Pixel (NBPP)	08	2 digits
Display Level (IDLVL)	001	3 characters - minimum value makes this base image
Attachment Level (IALVL)	000	Required 3 digit value since minimum display level.
Location (ILOC)	0000000000	10 characters

TABLE XX. Example image subheader of the base image - Continued.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
Image magnification (IMAG)	1.00	4 characters
User Defined Image Data Length (UDIDL)	00000	5 digits
Extended Subheader Data Length (IXSHDL)	00000	5 digits

TABLE XXI. Example image subheader of the first inset image.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (IM)	IM	2 characters
Image ID (IID)	Missing ID	10 characters
Image Date & Time (IDATIM)	25152559ZMAR93	14 characters
Target ID (TGTID)		17 spaces with 2 spaces
Image Title (ITITLE)	Zoomed Wind Tunnel	18 characters followed by 62 spaces - 80 characters total
Image Security Classification (ISCLAS)	U	1 character
Image Codewords (ISCODE)		40 spaces
Image Control and Handling (ISCTLH)		40 spaces
Image Releasing Instructions (ISREL)		40 spaces
Image Classification Authority (ISCAUT)		20 spaces
Image Security Control Number (ISCTLN)		20 spaces
Image Security Downgrade (ISDWNG)		6 spaces - no downgrade event
Encryption (ENCRYP)	0	Required default

TABLE XXI. Example image subheader of the first inset image - Continued.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
Image Source (ISORCE)	Unknown	42 spaces
Number of Significant Rows in image (NROWS)	00000656	8 characters
Number of Significant Columns in image (NCOLS)	00000656	8 characters
Pixel value type (PVTYPE)	INT	3 characters - interpret pixel values as integers
Image Representation (IREP)	MONO	4 characters followed by 4 spaces - grayscale imagery
Image Class (ICAT)	VIS	3 characters followed by 5 spaces - visible imagery
Actual Bits-Per-Pixel Per Band (ABPP)	08	2 digits
Pixel Justification (PJUST)	R	1 character
Image Coordinate System (ICORDS)	N	1 character - indicates no geo location coordinates
Number of Image Comments (NICOM)	0	1 digit
Image Compression (IC)	NC	2 characters - indicates uncompressed
Number of Bands (NBANDS)	1	1 digit
1st Band Representation (IREPBAND1)		2 spaces
1st Band Significance (ISUBCAT1)		6 spaces
1st Band Image Filter Condition (IFC1)	N	1 character - required default value
1st Band Standard Image Filter Code (IMFLT1)		3 spaces - reserved
1st Band Number of LUTS (NLUTS1)	0	1 character
Image Sync Code (ISYNC)	0	1 digit

TABLE XXI. Example image subheader of the first inset image - Continued.

NITF IMAGE SUBHEADER FIELD	FORMAT	COMMENT
Image Mode (IMODE)	B	1 character - B required for 1 band
Number of Blocks per Row (NBPR)	0001	4 digits
Number of Blocks per Column (NBPC)	0001	4 digits
Number of pixels Per Block Horizontal (NPPBH)	0656	4 digits
Number of pixels Per Block Vertical (NPPBV)	0656	4 digits
Number Bits Per Pixel (NBPP)	08	2 digits
Display Level (IDLVL)	002	3 digits
Attachment Level (IALVL)	001	3 digits
Location (ILOC)	0098000280	10 characters
Image Magnification (IMAG)	1.00	4 characters
User Defined Image Data Length (UDIDL)	00000	5 digits
Extended Subheader Data Length (IXSHDL)	00000	5 digits

TABLE XXII. Symbol subheader for the first symbol.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (SY)	SY	2
Symbol ID (SID)	0000000001	10
Symbol Name (SNAME)	CGM INSET FRAME	15 characters followed by 5 spaces - total 20 characters
Symbol Security Classification (SSCLAS)	U	1 character
Symbol Codewords (SSCODE)		40 spaces

TABLE XXII. Symbol subheader for the first symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Symbol Control and Handling (SSCTLH)		40 spaces
Symbol Releasing Instructions (SSREL)		40 spaces
Symbol Classification Authority (SSCAUT)		20 spaces
Symbol Security Control Number (SSCTLN)		20 spaces
Symbol Security Downgrade (SSDWNG)	999998	6 characters
Symbol Downgrading Event (SSDEVT)	This symbol will never need downgrading.	40 characters - full field
Encryption (ENCRYP)	0	Required default
Symbol Type (STYPE)	C	1 character - indicates CGM
Number of Lines Per Symbol (NLIPS)	0000	4 digits - does not apply to CGM
Number of Pixels Per Line (NPIXPL)	0000	4 digits - does not apply to CGM
Line Width (NWDTH)	0000	4 digits - does not apply to CGM
Number of Bits Per Pixel (NBPP)	0	1 digit - does not apply to CGM
Display Level (SDLVL)	003	3 digits
Attachment Level (SALVL)	002	3 digits
Symbol Location (SLOC)	0000000000	10 characters
Second Symbol Location (SLOC2)	0000000000	10 characters
Symbol Color (SCOLOR)		1 space - space required for CGM
Symbol Number (SNUM)	000000	6 characters
Symbol Rotation (SROT)	000	3 digits - value 0 required for CGM

TABLE XXII. Symbol subheader for the first symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Number of LUT Entries (NELUT)	000	3 digits
Extended Subheader Data Length (SXSHDL)	00000	5 digits

TABLE XXIII. Symbol subheader for the second symbol.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (SY)	SY	2
Symbol ID (SID)	0000000002	10
Symbol Name (SNAME)	CGM FRAME	9 characters followed by 11 spaces - total 20 characters
Symbol Security Classification (SSCLAS)	U	1 character
Symbol Codewords (SSCODE)		40 spaces
Symbol Control and Handling (SSCTLH)		40 spaces
Symbol Releasing Instructions (SSREL)		40 spaces
Symbol Classification Authority (SSCAUT)		20 spaces
Symbol Security Control Number (SSCTLN)		20 spaces
Symbol Security Downgrade (SSDWNG)	999998	6 characters
Symbol Downgrading Event (SSDEVT)	This symbol will never need downgrading.	40 characters - full field
Encryption (ENCRYP)	0	Required default
Symbol Type (STYPE)	C	1 character - indicates CGM
Number of Lines Per Symbol (NLIPS)	0000	4 digits - does not apply to CGM

TABLE XXIII. Symbol subheader for the second symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Number of Pixels Per Line (NPIXPL)	0000	4 digits - does not apply to CGM
Line Width (NWDTH)	0000	4 digits - does not apply to CGM
Number of Bits Per Pixel (NBPP)	0	1 digit - does not apply to CGM
Display Level (SDLVL)	004	3 digits
Attachment Level (SALVL)	001	3 digits
Symbol Location (SLOC)	0085300961	10 characters
Second Symbol Location (SLOC2)	0000000000	10 characters
Symbol Color (SCOLOR)		1 space - space required for CGM
Symbol Number (SNUM)	000000	6 characters
Symbol Rotation (SROT)	000	3 digits - value 0 required for CGM
Number of LUT Entries (NELUT)	000	3 digits
Extended Subheader Data Length (SXSHDL)	00000	5 digits

TABLE XXIV. Symbol subheader for the third symbol.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (SY)	SY	2 characters
Symbol ID (SID)	0000000003	10 characters
Symbol Name (SNAME)	BITMAPPED SYMBOL	17 characters followed by 3 spaces - total 20 characters
Symbol Security Classification (SSCLAS)	U	1 character
Symbol Codewords (SSCODE)		40 spaces

TABLE XXIV. Symbol subheader for the third symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Symbol Control and Handling (SSCTLH)		40 spaces
Symbol Releasing Instructions (SSREL)		40 spaces
Symbol Classification Authority (SSCAUT)		20 spaces
Symbol Security Control Number (SSCTLN)		20 spaces
Symbol Security Downgrade (SSDWNG)	999998	6 characters
Symbol Downgrading Event (SSDEVT)	This symbol will never need downgrading.	40 characters - full field
Encryption (ENCRYP)	0	Required default
Symbol Type (STYPE)	B	1 character - indicates bitmapped
Number of Lines Per Symbol (NLIPS)	0260	4 digits
Number of Pixels Per Line (NPIXPL)	0260	4 digits
Line Width (NWDTH)	0000	4 digits - does not apply to bitmapped symbols
Number of Bits Per Pixel (NBPP)	1	1 digit
Display Level (SDLVL)	005	3 digits
Attachment Level (SALVL)	001	3 digits
Symbol Location (SLOC)	0159501705	10 characters
Second Symbol Location (SLOC2)	0000000000	10 characters
Symbol Color (SCOLOR)	N	1 space
Symbol Number (SNUM)	000000	6 characters
Symbol Rotation (SROT)	000	3 digits - 000 required for bitmapped
Number of LUT Entries (NELUT)	000	3 digits

TABLE XXIV. Symbol subheader for the third symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Extended Subheader Data Length (SXSHDL)	00000	5 digits

TABLE XXV. Symbol subheader for the fourth symbol.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (SY)	SY	2
Symbol ID (SID)	0000000004	10
Symbol Name (SNAME)	HANGAR LABEL	12 characters followed by 8 spaces - total 20 characters
Symbol Security Classification (SSCLAS)	U	1 character
Symbol Codewords (SSCODE)		40 spaces
Symbol Control and Handling (SSCTLH)		40 spaces
Symbol Releasing Instructions (SSREL)		40 spaces
Symbol Classification Authority (SSCAUT)		20 spaces
Symbol Security Control Number (SSCTLN)		20 spaces
Symbol Security Downgrade (SSDWNG)	999998	6 characters
Symbol Downgrading Event (SSDEVT)	This symbol will never need downgrading.	40 characters - full field
Encryption (ENCRYP)	0	Required default
Symbol Type (STYPE)	C	1 character - indicates CGM
Number of Lines Per Symbol (NLIPS)	0000	4 digits - does not apply to CGM

TABLE XXV. Symbol subheader for the fourth symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Number of Pixels Per Line (NPIXPL)	0000	4 digits - does not apply to CGM
Line Width (NWDTH)	0000	4 digits - does not apply to CGM
Number of Bits Per Pixel (NBPP)	0	1 digit - does not apply to CGM
Display Level (SDLVL)	006	3 digits
Attachment Level (SALVL)	005	3 digits
Symbol Location (SLOC)	00070-0025	10 characters
Second Symbol Location (SLOC2)	0000000000	10 characters
Symbol Color (SCOLOR)		1 space - space required for CGM
Symbol Number (SNUM)	000000	6 characters
Symbol Rotation (SROT)	000	3 digits - value 0 required for CGM
Number of LUT Entries (NELUT)	000	3 digits
Extended Subheader Data Length (SXSHDL)	00000	5 digits

TABLE XXVI. Symbol subheader for the fifth symbol.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (SY)	SY	2
Symbol ID (SID)	0000000005	10
Symbol Name (SNAME)	RUNWAY LABEL	12 characters followed by 8 spaces - total 20 characters
Symbol Security Classification (SSCLAS)	U	1 character
Symbol Codewords (SSCODE)		40 spaces

TABLE XXVI. Symbol subheader for the fifth symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Symbol Control and Handling (SSCTLH)		40 spaces
Symbol Releasing Instructions (SSREL)		40 spaces
Symbol Classification Authority (SSCAUT)		20 spaces
Symbol Security Control Number (SSCTLN)		20 spaces
Symbol Security Downgrade (SSDWNG)	999998	6 characters
Symbol Downgrading Event (SSDEVT)	This symbol will never need downgrading.	40 characters - full field
Encryption (ENCRYP)	0	Required default
Symbol Type (STYPE)	C	1 character - indicates CGM
Number of Lines Per Symbol (NLIPS)	0000	4 digits - does not apply to CGM
Number of Pixels Per Line (NPIXPL)	0000	4 digits - does not apply to CGM
Line Width (NWDTH)	0000	4 digits - does not apply to CGM
Number of Bits Per Pixel (NBPP)	0	1 digit - does not apply to CGM
Display Level (SDLVL)	007	3 digits
Attachment Level (SALVL)	001	3 digits
Symbol Location (SLOC)	0228800865	10 characters
Second Symbol Location (SLOC2)	0000000000	10 characters
Symbol Color (SCOLOR)		1 space - space required for CGM
Symbol Number (SNUM)	000000	6 characters
Symbol Rotation (SROT)	000	3 digits - value 0 required for CGM

TABLE XXVI. Symbol subheader for the fifth symbol - Continued.

NITF SYMBOL SUBHEADER FIELD	FORMAT	COMMENT
Number of LUT Entries (NELUT)	000	3 digits
Extended Subheader Data Length (SXSHDL)	00000	5 digits

TABLE XXVII. Label subheader for the first label.

NITF LABEL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (LA)	LA	2 characters
Label ID (LID)	0000000001	10 characters
Label Security Classification (LSCLAS)	U	1 character
Label Codewords (LSCODE)		40 spaces
Label Control and Handling (LSCTLH)		40 spaces
Label Releasing Instructions (LSREL)		40 spaces
Label Classification Authority (LSCAUT)		20 spaces
Label Security Control Number (LSCTLN)		20 spaces
Label Security Downgrade (LSDWNG)	999998	6 characters
Label Downgrading Event (LSDEVT)	This label will never be downgraded.	36 characters followed by 4 spaces - 40 total characters
Encryption (ENCRYP)	0	required default
Label Font Style (LFS)		1 space
Label Cell Width (LCW)	56	2 digits
Label Cell Height (LCH)	75	2 digits
Label Display Level (LDLVL)	008	3 digits

TABLE XXVII. Label subheader for the first label - Continued.

NITF LABEL SUBHEADER FIELD	FORMAT	COMMENT
Attachment Level (LALVL)	001	3 digits
Label Location (LLOC)	0012200666	10 characters
Label Text Color (LTC)	1 1 1	3 bytes - each represented here in decimal
Label Background Color (LBC)	255 255 255	3 bytes - each represented here in decimal
Extended Subheader Data Length (LXSHDL)	00000	5 digits

TABLE XXVIII. Label subheader for the second label.

NITF LABEL SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (LA)	LA	2 characters
Label ID (LID)	0000000002	10 characters
Label Security Classification (LSCLAS)	U	1 character
Label Codewords (LSCODE)		40 spaces
Label Control and Handling (LSCTLH)		40 spaces
Label Releasing Instructions (LSREL)		40 spaces
Label Classification Authority (LSCAUT)		20 spaces
Label Security Control Number (LSCTLN)		20 spaces
Label Security Downgrade (LSDWNG)	999998	6 characters
Label Downgrading Event (LSDEVT)	This label will never be downgraded.	36 characters followed by 4 spaces - 40 total characters
Encryption (ENCRYP)	0	1 character - required default

TABLE XXVIII. Label subheader for the second label - Continued.

NITF LABEL SUBHEADER FIELD	FORMAT	COMMENT
Label Font Style (LFS)		1 space
Label Cell Width (LCW)	18	2 digits
Label Cell Height (LCH)	36	2 digits
Label Display Level (LDLVL)	009	3 digits
Attachment Level (LALVL)	003	3 digits
Label Location (LLOC)	-006500132	10 characters
Label Text Color (LTC)	1 1 1	3 bytes - each represented here in decimal
Label Background Color (LBC)	255 255 255	3 bytes - each represented here in decimal
Extended Subheader Data Length (LXSHDL)	00000	5 digits

TABLE XXIX. Text subheader for the text document.

NITF TEXT SUBHEADER FIELD	FORMAT	COMMENT
File Part Type (TE)	TE	2 characters
Text ID (TEXTID)	0000000001	10 characters
Text Date & Time (TXTDT)	27235536ZMAR93	14 characters
Text Title (TXTITL)	First sample text file.	22 characters followed by 58 spaces - 80 total characters
Text Security Classification (TSCLAS)	U	1 character
Text Codewords (TSCODE)		40 spaces
Text Control and Handling (TSCTLH)		40 spaces
Text Releasing Instructions (TSREL)		40 spaces
Text Classification Authority (TSCAUT)		20 spaces

TABLE XXIX. Text subheader for the text document - Continued.

NITF TEXT SUBHEADER FIELD	FORMAT	COMMENT
Text Security Control Number (TSCTLN)		20 spaces
Text Security Downgrade (TSDWNG)	999998	6 characters
Text Downgrading Event (TSDEVT)	This text will never be downgraded.	34 characters followed by 6 spaces - 40 total characters
Encryption (ENCRYP)	0	1 character - required default
Text Format (TXTFMT)	STA	3 characters
Extended Subheader Data Length (TXSHDL)	00000	5 digits

6.2.1 Explanation of the file header. The File Type and Version, NITF 02.00, is listed first. The next field contains the file's Compliance Level, in this case 04. A four character reserved field for the System Type, defaulted to blanks, appears next. An identification code containing ten characters for the station originating the primary information in the file is given next. The file origination date and time follow this and are given in ZULU time format. This is followed by the File Title, an optional field containing up to 80 characters of free form text. The title of the sample file contains less than 80 characters, and therefore, the remainder of the field is padded with blanks. The File Security Classification follows and contains one character. Several security-related optional fields and a conditional field follow. They are File Codewords, File Control Handling, File Releasing Instructions, File Classification Authority, File Security Control Number, File Security Downgrade, File Copy Number, and File Number of Copies. Notice that the File Security Downgrade is 999998, which is required in order for the following conditional field, File Security Downgrade Event, to be present. File Encryption is given a "0" indicating that the file is not encrypted. The originator's name and phone number are given next. These fields are optional and may be left blank. Then the length in bytes of the entire file is given, including all headers, subheaders, and data. This is followed by the length in bytes of the NITF file header. The Number of Images field contains the characters 002 to indicate two images are included in the file. This is followed by six characters to specify the length of the first image subheader, then ten characters for the length of the first image. The length of the second image subheader and the length of the second image follow. The next item in the file header is the Number of Symbols, which contains 005 to indicate that five symbols are present in the file. The next ten fields contain the Length of Symbol Subheader and Length of Symbol (four and six characters respectively) for the first through fifth symbol, one after the other. Next the Number of Labels is given as 002. The Length of Label Subheader (four characters) and Length of Label (three characters) follow for each of the labels. The field, Number of Text Files, is given as 005 and is followed by four characters specifying the length of the text subheader and five characters specifying the number of characters in the text segment for each fo the five text segments. The Number of Data Extension Segments and Number of Reserved Extension Segments fields are given as "000." Consequently, no segment subheader or segment lengths are given, since they are conditional fields. This completes the "roadmap" for separating the data subheaders from the actual data to follow. The next two fields in the header are the User Defined Header Data Length and the User Defined Header Data. User defined data could be used to include registered tagged record extensions that provide additional information about the file for which a field does not exist in the header. In this example, however, the length of the user defined header data is given as zero; therefore, the User Defined Header Data Field is omitted. The last two fields in the header are the Extended Header Data Length and the Extended Header Data. The length of the extended header is given as zero; therefore, the Extended Header Data field is omitted, indicating that no controlled tagged record extensions are included in the file.

6.2.2 Explanation of the image subheaders.

6.2.2.1 Explanation of the first image subheader (table XX). There are two images in this sample file. The first image has Display Level 001, which defines it as the base image. Its subheader is shown in Table XX. It is an unclassified, single band, single block, gray scale image with 8 bits per pixel and does not have an associated LUT. There are three associated comments. It is visible imagery, is not geo-located and has been stored in JPEG compressed format with a quality level of 1. It is located at the origin of the common coordinate system within which all the displayable file components are located. It is 2756 rows by 2194 columns. Figure 10 illustrates the image at three hundred pixels per inch.

FIGURE 10. Sample briefing board.

6.2.2.2 Explanation of the second image subheader (table XXI). This image is the second image in the file. As is the first image, this image is an 8 bit visible, gray scale image. It is much smaller (656 columns x 656 rows) and is not compressed. This image subheader is missing a number of conditional fields that were present in the first image. Notice that it has no downgrade event field (ISDEVT) and the ISDWNG field is blank rather than having the special 999998 code. Also, unlike the first image, it has no associated comment fields, indicated by the fact NICOM = 0. Since the image is uncompressed, the conditional COMRAT field is omitted. Since it is attached to the base image (IALVL = 001), the ILOC field reveals that this image is located with its upper left corner positioned at Row 980, Column 280 with respect to the upper left corner of the base image. Its lower left corner is at Row 1635, Column 935. Since it has a display level greater than that of the base image, it will obscure part of the base image when they are both displayed.

6.2.3 Explanation of the symbol subheaders.

6.2.3.1 Explanation of the first symbol subheader (table XXII). This symbol is a computer graphics metafile symbol. As a result, several fields have default content since they do not apply to CGMs (for example, NLIPS, NBPP, SCOLOR). The symbol consists of a shadowed frame for the inset image with a triangular pointer extending diagonally from near its upper right corner. The symbol is attached to the inset image, and therefore its location as recorded in SLOC is measured as an offset of the symbol's Virtual Display Coordinates (VDC) origin from the upper left corner of that image. The symbol's VDC origin is at the upper left corner of the image frame. Since the frame's outer boundary coincides with the edge of the inset image, the VDC origin is at an offset of 0 rows and 0 columns. This explains the contents of SLOC in table XXII.

6.2.3.2 Explanation of the second symbol subheader (table XXIII). The second symbol is also a CGM symbol. It is the simple rectangle surrounding the area on the base image that is expanded in the inset. It is attached to the base image. Therefore, its location as recorded in SLOC is measured as an offset from the upper left corner of the base image. The symbol's VDC origin is at its upper left corner, which is located at row 853, column 961 of the base image.

6.2.3.3 Explanation of the third symbol subheader (table XXIV). The third symbol is a bi-level, bitmapped symbol to be drawn interpreting 1 as black, 0 as transparent. It is attached to the base image. It is the U-shaped symbol bracketing the blimp hangar. Its location as recorded in SLOC is measured as an offset from the upper left corner of the base image, such as row 1595, Column 1705 of the base image (the symbol's upper left corner is its VDC origin).

6.2.3.4 Explanation of the fourth symbol subheader (table XXV). The fourth symbol is a CGM symbol. It is the "BLIMP HANGAR" text on a white rectangle with a black triangular pointer. It is attached to the third symbol, the U-shaped hangar bracket. Therefore, its location as recorded in SLOC is measured as an offset from the upper left corner of the hangar bracket. This symbol's VDC origin is at the junction of the left side of the black triangle with the white rectangle, which is located at row 1665, column 1580 of the base image. The offset from symbol three's VDC origin is therefore 70 rows, -25 columns.

6.2.3.5 Explanation of the fifth symbol subheader (table XXVI). The fifth symbol is a CGM symbol. It is the "RUNWAY" text on a white rectangle with a white downward pointing arrow. It is attached to the base image. Therefore, its location as recorded in SLOC is measured as an offset from the upper left corner of the base image. This symbol's VDC origin is at the upper left corner or the white rectangle, which is located at row 2288, column 865 of the base image.

6.2.4 Explanation of the label subheaders. There are two labels. Other than their actual text, they differ only in their attachment levels, location and text font size.

6.2.4.1 Explanation of the first label subheader. The first label is unclassified, attached to the base image and offset from its upper left corner by 122 rows and 666 columns. The label is black on a white background. It is the "MOFFETT FIELD" label at the top of composite image. The subheader for this label is shown in table XXVII.

6.2.4.2 Explanation of the second label subheader. The second label is unclassified, attached to the first symbol and offset from its upper left corner (its VDC origin) by -65 rows and 132 columns. Its upper left corner is coincident with the pixel 915,412 of the base image. The label is black on a white background. Its text is "WIND TUNNEL." The subheader us shown in table XXVII.

6.2.5 Explanation of the text subheaders. There are 5 text documents included in the file. Other than the text data they contain, they differ only in matters such as title, date-time of creation, and ID. Therefore, only the first is discussed, since the subheaders of all the rest are essentially the same.

6.2.5.1 Explanation of the first text subheader. The first text document is unclassified and was created on March 27, 1993 at 36 seconds past 23:55 hours. Its subheader is shown in table XXIX.

6.3 Effectivity summary. Effectivity 1 - Alternate display interpretation of NITF images. a. 5.5.1.1 Display of NITF images. ... Systems that require other interpretations of the image array (differing from the interpretation described in paragraph 5.5.1.1) shall manage these requirements within the file using (Effectivity 1).

6.4 Subject term (key word) listing.

Annotation, Imagery
Blocked Image Mask
BWC
Compression Algorithm
Compression, Bi-level
Compression, Imagery
DCT, Discrete Cosine Transform
Facsimile Compression
File Format
Graphics
Gray scale imagery
Group 3 facsimile
Huffman coding
Image
Image Compression
Image Dissemination
Image Transmission
Imagery, Bi-level
Overlay
Picture
Quantization Matrices
Raster
Secondary Imagery Dissemination Systems
SIDS
Symbols
Tag
Transparent Pixel
Transparent Pixel Mask

6.5 Changes from previous issue. Marginal notations are not used in this revision to identify changes with respect to the previous issue due to the extensiveness of the changes.

APPENDIX A

IMPLEMENTATION CONSIDERATIONS

10. GENERAL

10.1 Scope. This appendix is not a mandatory part of the standard. The information contained in it is explanatory and intended for guidance only.

10.2 NITF implementation guidelines. The National Imagery Transmission Format Standard (NITFS) has been developed to provide image exchange capabilities among computer systems of various designs and capabilities. This appendix will discuss general considerations pertinent to successful implementation of the NITF. Guidelines will be presented, and potential problems will be highlighted. The NITF preprocessor and postprocessor software, the software necessary to write and read an NITF file based on host files containing the data items to be included, are to be written by the user. The combination of the preprocessor and postprocessor hereafter will be referred to as the "NITF implementation." Preprocessing is sometimes called "packing," and postprocessing is called "unpacking." NITF implementation sample software is available through your point of contact.

20. APPLICABLE DOCUMENTS. This section is not applicable to this appendix.

30. DEFINITIONS

30.1 Definitions used in this appendix. For purposes of this appendix, the definitions are listed in Section 3.

40. GENERAL REQUIREMENTS

40.1 Scope of NITF implementation. NITF describes the format of images, symbols, labels and text within the NITF file only. It does not define the image or text requirements of the host system. The host system is responsible for the handling of unpacked image and text files, as well as image and text display capabilities.

40.2 Creating headers and subheaders. This standard specifies legal values for the header and subheader fields. The NITF preprocessor for any particular host system will be responsible for enforcing the field values as stated in this standard.

40.3 Character counts. The NITF uses explicit byte counts to delimit fields. No end-of-field characters are used. These byte counts are critical for the proper interpretation of an NITF file. The NITF preprocessor should compute these byte counts based on file contents to insure accuracy. All fields in the NITF header and subheaders must be present exactly as specified in the NITF header and subheader descriptions, and no additional fields may be inserted. The NITF uses various conditional fields whose presence is determined by previous fields and counts. If an expected conditional field is missing, the remainder of the file will be misinterpreted. A similar result will occur if a conditional field is inserted when it is not required. For these reasons, the item count fields are critical, and every effort must be made to ensure their accuracy. The NITF preprocessor should compute these item counts based on file contents whenever possible.

40.4 Data entry. To reduce any operator workload imposed by the preprocessor, each preprocessor, should provide for the automatic entry of data. Global default values for the particular NITF version should be inserted automatically in the file. System default values, such as the standard size parameters for a base image, also should be entered automatically by the preprocessor. Values that are known to the system, such as the time or the computed size of an overlay, also should be entered automatically.

40.5 User defined header and user defined image subheader data. Users may need to add additional data to an NITF file header or image subheader. To accommodate this requirement, user defined data fields are provided in the file header and image subheader. One potential use for the user defined image subheader data is to provide space for directly associating acquisition parameters with the image. Use of these fields requires insertion of tagged records that implement the extension as described in this standard. Before use, tags shall be registered with the NTB according to procedures available from the NTB. This procedure ensures that different users will not use the same tag to flag different extended data. It also provides for configuration management of tagged record formats where the extended data are expected to be used by a wide audience of users.

40.5.1 Handling the extended headers and subheaders. The NITF has made allowances for future enhancements by defining extended headers and subheaders, the contents of which are under configuration control (contact the NTB for information about format and use of these data). These fields should not be used except as provided for in documentation available from the NTB. These extended headers are composed of an extended header byte count and extended header data. The extended header count must be extracted by the software, and the appropriate number of extended header bytes must be read or bypassed. Five extended headers are in the current NITF format under configuration control. They are the Extended Header Data (XHD) in the NITF Header and the Extended Subheaders in the Image (IXSHD), Symbol (SXSHD), Label (LXSHD), and Text (TXSHD) Subheaders. The NITF also has made allowances for extended headers that are under user control by providing the User Defined Header Data (UDHD) field in the NITF Header and the User Defined Image Data (UDID) field in the Image Subheader. Use of these fields must be coordinated with the NTB by tag registration, but it is not under configuration management. Implementors are reminded that these extended headers also must be handled properly (skip over them if there are no means to interpret them properly).

40.6 Out of bounds fields. The file creator is responsible for ensuring that all NITF field values are within the bounds specified by the NITF document. An out-of-bounds value in an NITF field indicates that either an error occurred or that the sending station was not in full compliance with NITF.

40.7 Use of images in NITF. The NITF specifies a format for images contained within an NITF file only. An NITF implementation must be capable of translating this format to and from the host systems's local format. Some host systems have multiple formats for binary data. In these cases, the NITF implementation must use the appropriate host format to provide the necessary data exchange services with other system packages. When imagery data of less than M bits-per-pixel is displayed on an M-bit (2^M gray shades) display device, it must be transformed into the dynamic range of the device. One way to do this is to modify the LUTs of the display device. However, if M-bit and less than M-bit imagery is displayed simultaneously, the M-bit image will appear distorted. The recommended method is to convert the less than M-bit imagery into M-bit imagery, then use the standard LUTs. The following equation will transform a less than M-bit pixel into and M-bit pixel:

N = number of bits-per-pixel
P_N = N-bit pixel value
P_M = M-bit pixel value

$$P_M = \frac{2^M - 1}{2^N - 1} P_N$$

40.8 Use of labels in NITF. The NITF does not attempt to identify standard fonts for labels. Such a task is beyond the scope of this effort. The label fields are intended to provide a quick and efficient method for exchanging textual information. Therefore, labels will be displayed in the host system's native font. If a particular font and/or font size relative to the image is necessary, a bit-mapped symbol may be substituted for the label. A label with multiple lines of text may be represented as a single label by separating the lines with carriage return/line feed pairs imbedded in the label text. This convention may require translation to or from the local system's use of line feeds and carriage return characters.

40.9 Use of text files in the NITF. The text format field is provided to help the reader of the file determine how to interpret the text data received. The file reader is responsible for interpreting the various text formats. Format designations explicitly supported by the NITF are as follows:

a. USMTF. USMTF is the United States Message Text Format.

b. NITF ASCII. NITF ASCII is a special format by the NTB to provide a common format for all NITF implementations. The ASCII code shall be represented as depicted on figure A-1. This is the ASCII code represented in ANSI X3.4-1986. The ASCII codes shall be seven bits, a_1 thru a_7 with an eighth bit added. The eighth bit, a_8, shall be set to 0. A_8 shall be the Most Significant Bit (MSB), and a_1 shall be the Least Significant Bit (LSB). It is intended to provide for simple communications among NITFS stations. The NITF ASCII format is comprised of the following ASCII characters (all numbers are decimal): Line Feed (10), Form Feed (12), Carriage Return (13), and space (32) through tilde (126). This set includes all the alphanumeric characters as well as all commonly used punctuation characters. All lines within an NITF ASCII file will be separated by carriage return/line feed pairs. It is the responsibility of the local system to translate these pairs into the local format. NITF ASCII has no standard line length. The host system must be capable of processing lines that are longer than the local standard.

c. Other. "Other" will allow all eight-bit codes to be transmitted. Different systems interpret these codes for various purposes. This format should be restricted to uses where the receiving and transmitting stations have agreed beforehand what the format represents.

40.10 NITF alphanumeric. Throughout this standard the term "alphanumeric" is applied to include all NITF ASCII characters except Line Feed (10), Form Feed (12), and Carriage Return (13). Common punctuation marks are legitimate "alphanumeric" characters.

40.11 File system constraints. An NITF file is presented as a stream of contiguous bytes. This format may not be suitable for some file systems. The translation of files to and from the local file format for a system should be examined for potential incompatibilities before an implementation is attempted.

40.12 Security considerations. An NITF file contains sufficient security information in the file header, image, symbol, and label subheaders to allow implementors to meet virtually any security requirement requiring display of classification data. Exact security information handling requirements generally are specified by appropriate accreditation authorities or specific user requirements. It is suggested that implementors extract the classification data from one or more of the header/subheaders and ensure that the information always is displayed whenever the pertinent part of the NITF file is displayed.

Bits				b8 →	0	0	0	0	0	0	0	0
				b7 →	0	0	0	0	1	1	1	1
				b6 →	0	0	1	1	0	0	1	1
				b5 →	0	1	0	1	0	1	0	1
b_4 ↓	b_3 ↓	b_2 ↓	b_1 ↓	Column → / Row ↓	0	1	2	3	4	5	6	7
0	0	0	0	0	NUL	DLE	SP	0	@	P	'	p
0	0	0	1	1	SOH	DC1	!	1	A	Q	a	q
0	0	1	0	2	STX	DC2	"	2	B	R	b	r
0	0	1	1	3	ETX	DC3	#	3	C	S	c	s
0	1	0	0	4	EOT	DC4	$	4	D	T	d	t
0	1	0	1	5	ENQ	NAK	%	5	E	U	e	u
0	1	1	0	6	ACK	SYN	&	6	F	V	f	v
0	1	1	1	7	BEL	ETB	'	7	G	W	g	w
1	0	0	0	8	BS	CAN	(	8	H	X	h	x
1	0	0	1	9	HT	EM	)	9	I	Y	i	y
1	0	1	0	10	LF	SUB	*	:	J	Z	j	z
1	0	1	1	11	VT	ESC	+	;	K	[	k	{
1	1	0	0	12	FF	FS	,	<	L	\	l	¦
1	1	0	1	13	CR	GS	-	=	M	]	m	}
1	1	1	0	14	SO	RS	.	>	N	^	n	~
1	1	1	1	15	SI	US	/	?	O	_	o	DEL

b_1 = Least Significant Bit (LSB)

b_8 = Most Significant Bit (MSB)

FIGURE A-1. ASCII eight-bit characters.

APPENDIX B

TAGGED RECORD EXTENSIONS

10. Scope. This appendix is a mandatory part of the standard. The information contained herein is intended for compliance.

20. Applicable documents. Information on applicable documents associated with a specific tagged record extension will be included in the definition section for that tag.

30. NITF tagged record extensions. This appendix contains information about the tagged record extensions (tags) used for NITF 2.0 files. All tag names are registered with the NITFS Technical Board. A listing of registered names is provided in 30.1. This appendix establishes the format and provides a detailed description of the data and data formats for the controlled tagged record extensions for NITF 2.0 files. Definitions for controlled tags that have been approved by the NTB are provided in 30.2. These controlled tag definitions are under formal configuration management by the NTB.

30.1 List of registered tag names. The following is the list of tag names that have been registered with the NTB. This list will be updated periodically as new tag names are registered.

a. ASYM
b. ATTRIB
c. BLKOFF
d. ESD001
e. ESD002
f. ESD003
g. ESD004
h. ESD005
i. ESD006
j. ESD007
k. ESD008
l. ESD009
m. ESD102
n. ESD103
o. ESD104
p. ESD105
q. ESD106
r. ESD107
s. ESD108
t. ESD109
u. IMS__
v. NOBASE
w. OFFSET
x. PIXS01 through PIXS99
y. RPC_

z. RPFDES
aa. RPFHDR
ab. RPFIMG
ac. RULER__

30.2 Controlled tag definitions. The following controlled tag definitions have been approved by the NTB and are under formal configuration control.

30.2.1 ESD001 tagged record extension description. This definition establishes the format and provides a detailed description of the data and data format for the controlled tagged record extension ESD001 to the NITF 2.0. The ESD001 tagged record supports inclusion in a NITF 2.0 file of basic Exploitation Support Data (ESD) for earth images. The valid data content for each field of the record, as well as information on the proper interpretation of the information is included.

30.2.1.1 Applicable documents for ESD001. The following documents of exact issue shown, form a part of this definition to the extent specified herein.

WGS December 84	Department of Defense World Geodetic System 1984 (WGS-84) Part II Parameters and Graphics for Practical Applications of WGS-84, 31 December 1984.	
	Reference pages 5-241-1	Grids and references
	Reference pages 5-241-8	Universal Transverse Mercator Grid

30.2.1.2 Format description. Table B-1 defines the format for the controlled tagged record extension to the NITF 2.0 bearing tag ESD001. ESD001 is intended for use with earth images when the user requires the image's ground location and the image scale. The information included in this record supports basic mensuration functions such as:

a. Determination of the ground location for a point in the image

b. Determination of the ground distance between two points in the image

c. Determination of the height of an object spanning two shadow points in the image

d. Determination of true north in the image

e. Accuracies for the above mensuration functions

TABLE B-1. ESD001 field formats.

FIELD NAME	BYTES	DESCRIPTION	FORMAT ASCII	UNITS	RANGE
CETAG	6	UNIQUE EXTENSION IDENTIFIER	ESD001	Not applicable	Not applicable
CEL	5	LENGTH OF ENTIRE TAGGED RECORD	00160	BYTES	160
		THE FOLLOWING FIELDS DEFINE CEDATA			
POINT C LATITUDE	14	LATITUDE AT POINT C DEGREES MINUTES SECONDS FRAC OF SECONDS ORIENTATION	DD:MM:SS.XXXXA DD MM SS XXXX A	DEG MIN SEC SEC FRAC Not applicable	00-90 00-59 00-59 0000-9999 N,S
POINT C LATITUDE ACCURACY	13	LATITUDE ACCURACY AT POINT C DEGREES MINUTES SECONDS FRAC OF SECONDS	DD:MM:SS.XXXX DD MM SS XXXX	DEG MIN SEC SEC FRAC	00-90 00-59 00-59 0000-9999 BLANKS FOR N/A
POINT C LONGITUDE	15	LONGITUDE AT POINT C DEGREES MINUTES SECONDS FRAC OF SECONDS ORIENTATION	DDD:MM:SS.XXXX A DDD MM SS XXXX A	DEG MIN SEC SEC FRAC Not applicable	00-180 00-59 00-59 0000-9999 E,W
POINT C LONGITUDE ACCURACY	13	LONGITUDE ACCURACY AT POINT C DEGREES MINUTES SECONDS FRAC OF SECONDS	DD:MM:SS.XXXX DD MM SS XXXX	DEG MIN SEC SEC FRAC	00-90 00-59 00-59 0000-9999 BLANKS FOR N/A

TABLE B-1. ESD001 field formats - Continued.

FIELD NAME	BYTES	DESCRIPTION	FORMAT ASCII	UNITS	RANGE
LINE AC ORIENTATION	14	ORIENTATION OF LINE AC FROM TRUE NORTH DEGREES MINUTES SECONDS FRAC OF SECONDS	DDD:MM:SS.XXXX DDD MM SS XXXX	 DEG MIN SEC SEC FRAC	 000-360 00-59 00-59 0000-9999
LINE AC ORIENTATION ACCURACY	13	ORIENTATION ACCURACY OF LINE AC DEGREES MINUTES SECONDS FRAC OF SECONDS	DD:MM:SS.XXXX DD MM SS XXXX	 DEG MIN SEC SEC FRAC	 00-90 00-59 00-59 0000-9999 BLANKS FOR N/A
LINE BC ORIENTATION	14	ORIENTATION OF LINE BC FROM TRUE NORTH DEGREES MINUTES SECONDS FRAC OF SECONDS	DDD:MM:SS.XXXX DDD MM SS XXXX	 DEG MIN SEC SEC FRAC	 000-360 00-59 00-59 0000-9999
LINE BC ORIENTATION ACCURACY	13	ORIENTATION ACCURACY OF LINE BC DEGREES MINUTES SECONDS FRAC OF SECONDS	DD:MM:SS.XXXX DD MM SS XXXX	 DEG MIN SEC SEC FRAC	 00-90 00-59 00-59 0000-9999 BLANKS FOR N/A
LINE AC LENGTH	10	LENGTH OF LINE AC	XXXX.XXXXX	NMI	0000.00001 TO 9999.99999
LINE AC LENGTH ACCURACY	8	LENGTH ACCURACY OF LINE AC	XX.XXXXX	NMI	00.00000 TO 99.99999 BLANKS FOR N/A
LINE BC LENGTH	10	LENGTH OF LINE BC	XXXX.XXXXX	NMI	0000.00000 TO 9999.99999

TABLE B-1. ESD001 field formats - Continued.

FIELD NAME	BYTES	DESCRIPTION	FORMAT ASCII	UNITS	RANGE
LINE BC LENGTH ACCURACY	8	LENGTH ACCURACY OF LINE BC	XX.XXXXX	NMI	00.00000 TO 99.99999 BLANKS FOR N/A
PIXEL AVERAGE	9	AVERAGE DISTANCE BETWEEN PIXELS AT CENTER	XXXXX.XXX	FT	00000.001 TO 99999.999
SUN ELEVATION ANGLE	6	SUN ELEVATION ANGLE FROM THE CENTER DEGREES FRAC OF DEGREES	±DD.XX DD XX	 DEG DEG FRAC	 ±00-±90 00-99 BLANKS FOR N/A

30.2.1.2.1 Figure B-1 illustrates the relationship between specific points, A, B, and C in the image with respect to Table B-1. Point C is the theoretical image center. If the image height (M) and width (N) are both odd then Point C will coincide with the pixel at row (M-1)/2 and column (n-1)/2, otherwise Point C will occur between pixels at the center of the image. Points A and B are the midpoints of the top edge and right edge, respectively. If N is odd, then Point A will coincide with the pixel at row 0 and column (N-1)/2, otherwise Point A will occur between pixels. If M is odd then Point B will coincide with the pixel at row (m-1)/2 and column N-1, otherwise Point B will occur between pixels. Note, angle BAC is a right angle on the image with respect to pixels but may not form a right angle for the image's ground locations. The orientations and lengths of lines AB and BC determine the skewness of the image pixels with respect to ground locations.

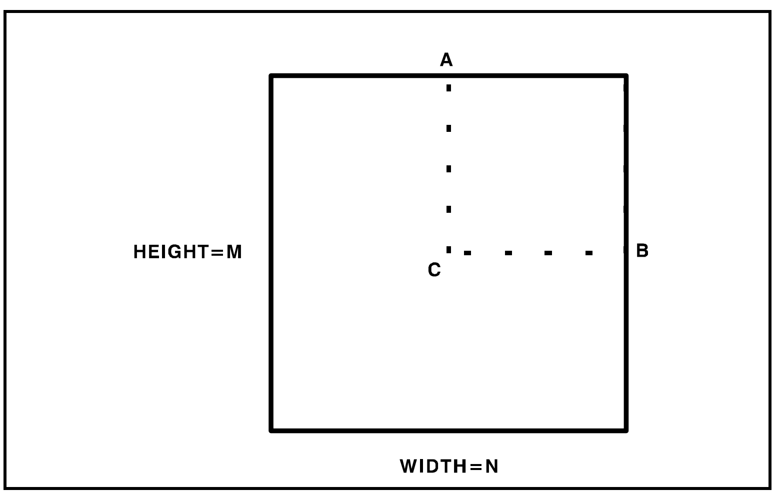


FIGURE B-1. Location reference points.

30.2.1.2.2 The following comments apply to the above table and figure:

a. The following abbreviations are used:

DEG	degrees
MIN	minutes
SEC	seconds
SEC FRAC	fractions of second
DEG FRAC	fractions of degree
NMI	nautical mile
FT	feet
N/A	not available

b. ACCURACY values shall be specified as two standard deviations of error, assuming a Normal distribution of error. Thus, the meaning of an accuracy value is that the value in the field to which the accuracy applies, plus or minus the accuracy value, defines an interval that has 95 percent probability of containing the correct value for that field.

c. ACCURACY values are blank filled when not available (N/A).

d. SUN ELEVATION ANGLE is measured with respect to the ground plane tangent at Point C and is blank filled when not available (N/A).

e. All coordinates are represented using WGS-84.

f. Orientation is measured clockwise from true north in the ground plane tangent at Point C.

30.2.2 OFFSET tagged record extension description. This definition establishes the format and provides a detailed description of the data and data format for the controlled tagged record extension OFFSET to the NITF 2.0. This extension defines the offset of the first pixel of an NITF 2.0 image from the first pixel of the full image described by the accompanying support data. If the NITF 2.0 image is blocked differently from the full image, or is not aligned to the full image block structure, this extension allows the NITF 2.0 image to be located relative to the full image, such that the support data can be properly utilized.

30.2.2.1 Format description. Table B-2 defines the format for the controlled tagged record extension to the NITF bearing tag OFFSET.

TABLE B-2. OFFSET format.

(R) = required, (O) = optional, and (C) = conditional

FIELD	DESCRIPTION	SIZE	FORMAT VALUE	TYPE
CETAG	Tag Record Identifier	1-6	OFFSET	R
CEL	Tag Data Field Length	5	00016	R
LINE	Align-Scan Offset of first pixel	8	0-99999999	R
SAMPLE	Cross-Scan Offset of first pixel	8	0-99999999	R

CONCLUDING MATERIAL

Custodians:
Army - SC
Navy - OM
Air Force - 02
Misc - DC

Review activities:
OASD - SO, DO, HP, IR
Army - AM, AR, MI, TM, MD, CE, SC, IE, ET, AC, PT
DLA - DH
Misc - NS, MP, DI, NA,CI

Civil agency coordinating activities:
USDA - AFS, APS
COM - NIST
DOE
EPA
GPO
HHS - NIH
DOI - BLM, GES, MIN
DOT - CGCT

Preparing activity:
Misc - DC

Agent:
Not applicable

(Project INST-0002)

STANDARDIZATION DOCUMENT IMPROVEMENT PROPOSAL

INSTRUCTIONS

1. The preparing activity must complete blocks 1,2, 3, and 8. In block 1, both the document number and revision letter should be given.
2. The submitter of this form must complete blocks 4, 5, 6, and 7.
3. The preparing activity must provide a reply within 30 days from receipt of the form.

NOTE: This form may not be used to request copies of documents, nor to request waivers, or clarification of requirements on current contracts. Comments submitted on this form do not constitute or imply authorization to waive any portion of the referenced document(s) or to amend contractual requirements.

I RECOMMEND A CHANGE:	1. DOCUMENT NUMBER MIL-STD-2500A	2. DOCUMENT DATE (YYMMDD) 941012

3. DOCUMENT TITLE NATIONAL IMAGERY TRANSMISSION FORMAT (VERSION 2.0)

4. NATURE OF CHANGE *(Identify paragraph number and include proposed rewrite, if possible. Attach extra sheets as needed.)*

5. REASON FOR RECOMMENDATION

6. SUBMITTER

a. NAME *(Last, First, Middle Initial)*	b. ORGANIZATION	
c. ADDRESS *(Include Zip Code)*	d. TELEPHONE *(Include Area Code)* (1) Commercial (2) AUTOVON *(If applicable)*	7. DATE SUBMITTED *(YYMMDD)*

8. PREPARING ACTIVITY DEFENSE INFORMATION SYSTEMS AGENCY (DISA)

a. NAME DISA/JIEO/CFS	b. TELEPHONE *(Include Area Code)* (1) Commercial (2) AUTOVON
c. ADDRESS *(Include Zip Code)* Parkridge III 10701 Parkridge Boulevard Reston, VA 22091-4398	**IF YOU DO NOT RECEIVE A REPLY WITHIN 45 DAYS, CONTACT:** Defense Quality and Standardization Office 5203 Leesburg Pike, Suite 1403, Falls Church, VA 22041-3466 Telephone (703) 756-2340 AUTOVON 289-2340