NATIONAL GEOSPATIAL-INTELLIGENCE AGENCY

NGA.STND.0024-2_1.0
2011-08-01

NGA STANDARDIZATION DOCUMENT

SENSOR INDEPENDENT COMPLEX DATA (SICD)

Volume 2

File Format Description Document

Specification of the placement of SICD data products
in the allowed image file formats.

(2011-08-01)

Version 1.0

NATIONAL CENTER FOR GEOSPATIAL INTELLIGENCE STANDARDS

CONTACTS

The following points of contact are provided for assistance in understanding the contents of this implementation profile.

NGA/AEX
P-001
12310 Sunrise Valley Drive
Reston, VA 20191-3449
(703) 755-5951

NGA/NCGIS
P-106
12310 Sunrise Valley Drive
Reston, VA 20191-3449
(703) 814-4564
NCGIS-mail@nga.mil

CHANGE LOG

Date	Version	Description	DR/CA	Developer
2011-08-11	1.0	Initial publication.		

TBR/TBD LOG

Page Number	TBD/TBR	Description	Date Addressed

FOREWORD

The suite of Sensor Independent Complex Data (SICD) standardization documents describe the implementation of SICD products for complex image data generated by Synthetic Aperture Radar (SAR) systems and their data processing elements.

A SAR complex image is an intermediate data product. The real utility is in the products and measurements that may be derived from it. The quality of the pixel array (resolution, SNR, etc.), along with the set of metadata provided, are critical in generating the derived products. The "sensor independence" of the SICD product refers to the ability of the allowed pixel array and metadata options to accurately describe the image products from many sensors and data processing systems. Sensor independence does NOT mean that all products have the same format for the pixel array or the same set of metadata parameters.

The SICD documentation has been organized into three volumes and a set of XML implementation artifacts. The three volumes are summarized below. The collection of SICD XML artifacts includes the schema documents that define the correct implementation of the XML metadata document included in a given product.

Volume 1 Design & Implementation Description Document

Contains the description needed by producers of SAR complex image products to design a SICD product and the set of metadata that describe it.

Volume 2 File Format Description Document

Defines the placement of SICD data products in the allowed image file formats. Also provides the description needed by users of SICD products to read and properly extract the SICD data components from a SICD product file.

Volume 3 Image Projections Description Document

Describes the SICD sensor model and the correct projections from image location to ground point and from ground point to image location for all SICD products.

A companion suite of standardization documents, collectively known as Sensor Independent Derived Data (SIDD), describes standardized products and measurements that may be derived from SICD.

The SICD and SIDD documentation and associated XML artifacts are available on the National System for Geospatial-Intelligence (NSG) Standards Registry (https://nsgreg.nga.mil).

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1 Introduction

1.1 Scope

The Sensor Independent Complex Data (SICD) File Format Description Document provides detailed specifications for how SICD products are placed in the allowed SICD container files. A SICD product is divided into a pair of data components. The data components include the complex image pixel data and the metadata that describe it. See Section 2.

This document defines how the SICD data components are mapped to the components of the allowed container files. A typical container file format includes required file components (e.g. a file header) as well as optional and conditional file components. How the required and optional (if any) file components are populated is defined. Provided is the description needed by the producers of SICD products to properly place the SICD data components into the product file. Also provided is the description needed by users of SICD products to read and properly extract the SICD data components from a SICD product file.

Currently, only a single container file format has been selected for SICD product files. The format is the National Imagery Transmission Format, Version 2.1 (referred to as NITF 2.1). The NITF 2.1 standard is described in MIL-STD-2500C, National Imagery Transmission Format, Version 2.1, dated 01 May 2006.

The National System for Geospatial Intelligence Metadata Foundation (NMF) – Part 1: Conceptual Schema Profile defines the *conceptual* schema profile for specifying geospatial metadata in and for the NSG. For purposes of this specification, each SICD NITF2.1 file is considered to be a separate resource that can be described per the NMF conceptual schema using information available within the SICD NITF2.1 file, and from known (a priori) information available to the process used to create the NMF-based description (e.g. information about the metadata itself and its originator that is not available within the resource being described).

1.2 Applicable Documents

The documents listed below are referenced in this document. Users of this document should investigate recent editions and change notices of the documents listed below.

Table 1-1 Government Documents and Publications

Number	Title & Website	Date
MIL-STD-2500C	National Imagery Transmission Format, Version 2.1 http://assist.daps.dla.mil/quicksearch/basic_profile.cfm?ident_number=112606	01 May 2006
STANAG 4545 NSIF Registry of Approved Data Extension Segments (DES) XML_DATA_CONTENT	XML_DATA_CONTENT Data Extension Segment Specification for use with the NATO Secondary Imagery Format (NSIF) and the National Imagery Transmission Format Standard (NITFS). http://www.nato.int/structur/AC/224/standard/4545/4545_documents/XML_DATA_CONTENT%20DES%20Final.pdf	30 June 2010
NGA.STND.0012_2.0	National System for Geospatial Intelligence metadata Foundation (NMF) - Part 1: Conceptual Schema Profile, Version 2.0 https://nsgreg.nga.mil/doc/view?i=2142	16 December 2010
NGA.STND.0024-1_1.0	Sensor Independent Complex Data, Volume 1, Design & Implementation Description Document, Version 1.0 (Draft)	Draft
NGA.STND.0024-3_1.0	Sensor Independent Complex Data, Volume 3, Image Projections Description Document, Version 1.0 (Draft)	Draft

2 SICD Data Components

2.1 SICD Product Data Components

A SICD product consists of a pair of data components. The first component is a block of complex image pixel data. The second component is a block of metadata expressed using the eXtensible Markup Language (XML). The metadata contains the selected set of parameters that describe the imaging collection and the data processing that formed the image. The SICD data components are described in detail in the Sensor Independent Complex Data Design & Implementation Description Document. The SICD components are briefly described below.

The SICD image pixel data block is a single two-dimensional array of complex numbers. The image pixel array contains NumRows rows by NumCols columns. The SICD image array size limits have not been established. The following are estimated values that are not expected to be exceeded in any current or foreseeable products.

NumRows:	1 to 1,000,000
NumCols:	1 to 1,000,000
Number of Pixels:	1 to 100,000,000,000 (e.g. 100,000 x 1,000,000)

The NITF 2.1 SICD container file format discussed in Section 3 will accommodate images of this size.

The SICD image pixel data may be stored in one of three allowed formats. The selected pixel format is specified by metadata parameter PixelType. The allowed pixel types are as follows.

PixelType = RE32F_IM32F

Each pixel is stored as a pair of numbers that represent the real and imaginary components. Each component is stored in a 32-bit IEEE floating point format (4 bytes per component, 8 bytes per pixel).

PixelType = RE16I_IM16I

Each pixel is stored as a pair of numbers that represent the real and imaginary components. Each component is stored in a 16-bit signed integer in 2's complement format (2 bytes per component, 4 bytes per pixel).

PixelType = AMP8I_PHS8I

Each pixel is stored as a pair of numbers that represent the amplitude and phase components. Each component is stored in an 8-bit unsigned integer (1 byte per component, 2 bytes per pixel).

The real and imaginary or amplitude and phase components of a given pixel are interleaved such that they are stored in adjacent bytes. For real and imaginary components, the real component is stored first. For amplitude and phase components, the amplitude component is stored first. For a given image, all pixels are stored in the same pixel format. The binary numbers are stored with the Most Significant Byte (MSB) first (referred to as "Big Endian"). The Big Endian byte order is required for placement in a NITF 2.1 container file.

The adjacent pixels within a row are stored in adjacent bytes within the file. The number of bytes per pixel is denoted by parameter BytesPerPixel (where BytesPerPixel = 2, 4 or 8). The NumCols pixels of the first row in the image array are stored in the first bytes in the file (BytesPerPixel x NumCols bytes). The total size of the image array is BytesPerPixel x NumCols x NumRows bytes.

The SICD XML metadata block contains the set of metadata selected to support the image. The XML block includes parameters describing the radar collection geometry, key image formation processing parameters and the parameters needed for correct exploitation. The size limits of the XML metadata block have not been established but the following are estimated values not expected to be exceeded.

XML Metadata Size: 10 KB to 10 MB

Contained within the SICD XML metadata block are the parameters used to place the SICD data components into the file components of the selected SICD container file. This set of parameters is referred to as the Container Placement Parameters. These parameters are used to populate the container file header and sub-headers and to place the data components into the file components. The Container Placement Parameters are listed in Table 2-1 below. All Container Placement Parameters are required parameters.

The SICD Container Placement Parameters have been selected based upon the NITF 2.1 container file format implementation. If additional SICD container file formats are added, the set of Container Placement Parameters may need to be expanded. The motivation for identifying the Container Placement Parameters as a set is to make file writer development easier.

2.2 Image Arrays: Full Image & Sub-Image

The image pixel array contained in a SICD product may be the initial image array as produced by the image formation processing system or a sub-image extracted from the initial image. The initial image is referred to as the "full image". A sub-image, sometimes referred to as an "image chip", is a single two-dimensional portion of the full image. The full image pixel array is NumRowsFI rows by NumColsFI columns. An example of a full image array is shown in Figure 2-1. The rows of the full image pixel array are indexed as follows.

Full image row index: row = 0, 1, 2, . . . , NumRowsFI – 1.

Full image column index: col = 0, 1, 2, . . . , NumColsFI – 1.

Image pixel indices (row, col) are referred to as the "global" image indices. The first row of the full image is indexed row = 0. The first column of the full image is indexed col = 0. Parameters NumRowsFI and NumColsFI are contained in the XML metadata for all image products.

Image size parameters NumRows and NumCols always denote the size of the image pixel array contained in the product file. See Figure 2-2. The rows and columns of the image pixel array in the product are indexed as follows.

Product row index: row_P = 0, 1, 2, . . . , NumRows – 1.

Product column index: col_P = 0, 1, 2, . . . , NumCols – 1.

Array indices (row_p, col_p) are useful in the computations needed to place the array in the container file. When the full image is placed in the product, product row and column indices are equal to the global row and column indices. Array size parameters NumRows and NumCols are used to place the image pixel array into the container file. Information about the Full Image array is not used.

The correct exploitation of a sub-image is dependent upon the relative position of the sub-image within the full image array. An example sub-image array and the full image array from which it was extracted are shown in Figure 2-3. The first row of the sub-image (row_P = 0) is extracted from row = FirstRow of the full image. The first column of the sub-image (col_P = 0) is extracted from col = FirstCol of the full image. FirstRow and FirstCol are contained in the XML metadata for all image products. For a full image product, parameters FirstRow = 0 and FirstCol = 0.

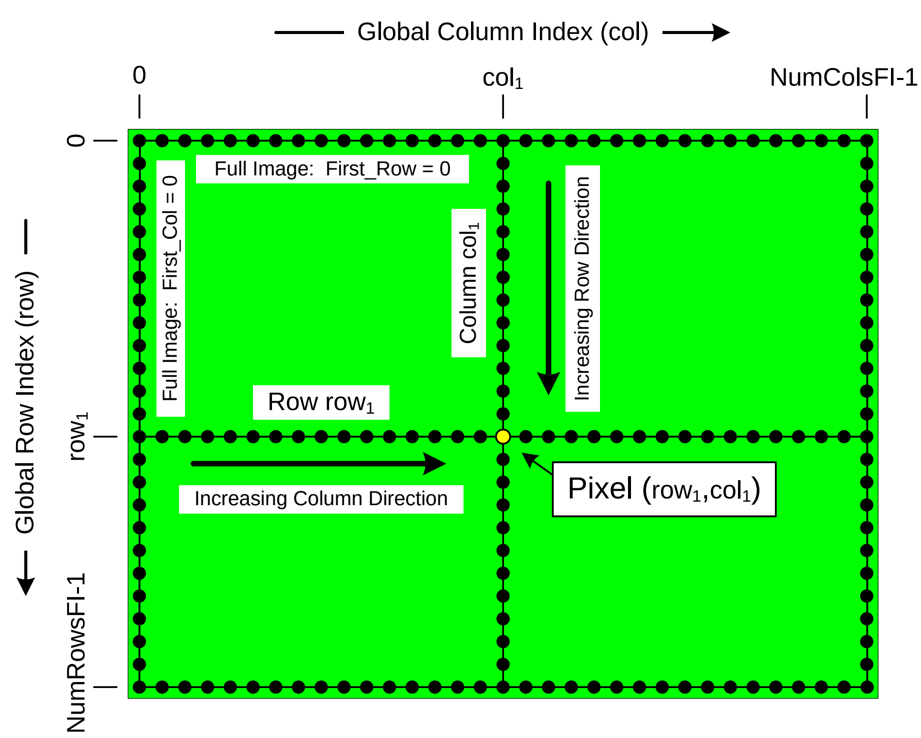

Figure 2-1 Full Image Pixel Array

Global image indices (row,col) are used to address the elements of the SICD pixel array .

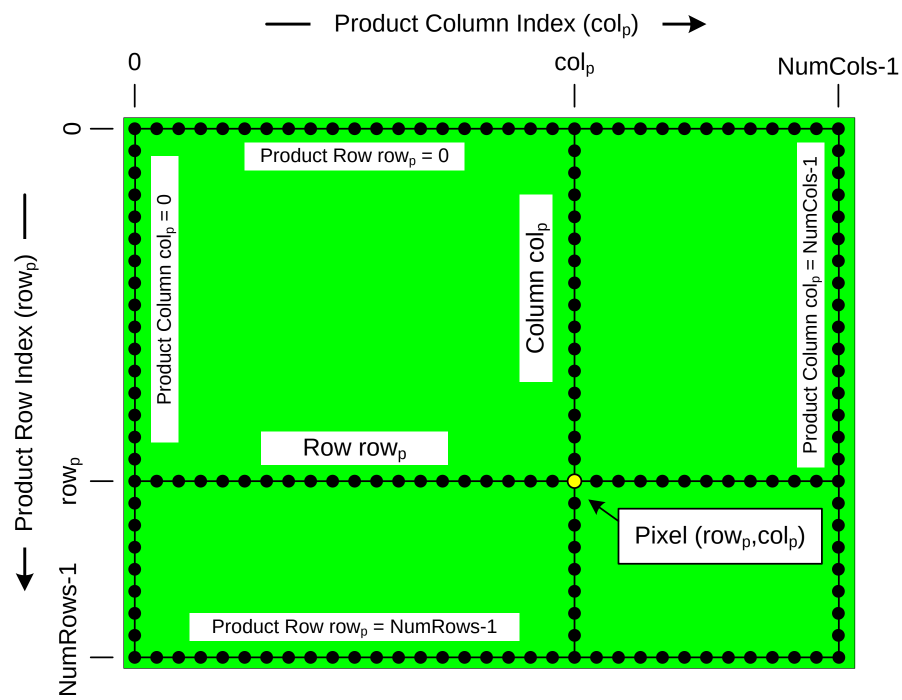

Figure 2-2 Image Pixel Array In A Product

Pixel array size in a product always specified by NumRows and NumCols.

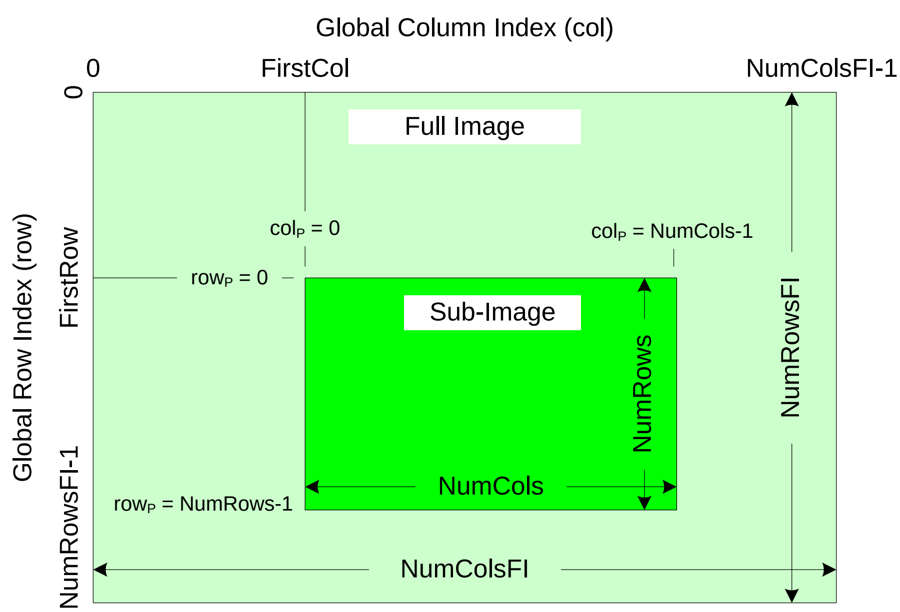

Figure 2-3 Sub-Image Pixel Array In A Product

Sub-image pixel array location in the full image array specified by parameters First_Row & First_Col.

2.3 SICD Container Placement Parameters

Table 2-1 SICD Container Placement Parameters

Parameter	XML Field Name	Req / Opt	Type	Description
CoreName	SICD.CollectionInfo.CoreName	R	TXT	Collection & imaging data set identifier per Program Specific Implementation Document.
CollectorName	SICD.CollectionInfo.CollectorName	R	TXT	Radar platform identifier. For Bistatic collections, list the Receive platform.
CollectStart	SICD.Timeline.CollectStart	R	XDT	Collection date and start time (UTC).
Classification	SICD.CollectionInfo.Classification	R	TXT	Contains the human-readable banner. Contains classification, file control & handling, file releasing, and/or proprietary markings. Specified per Program Specific Implementation Document. Default value: “UNCLASSIFIED”.
PixelType	SICD.ImageData.PixelType	R	ENU	Indicates the type and binary format used for individual pixels. PixelType = RE32F_IM32F ⇔ Real & Imaginary stored in 32 bit floating point format. PixelType = RE16I_IM16I⇔ Real & Imaginary stored in 16 bit signed integer 2’s complement format. PixelType = AMP8I_PHS8I⇔ Amplitude & Phase stored in 8 bit unsigned integer format.
NumRows Range of values: 1 to 1,000,000.	SICD.ImageData.NumRows	R	INT	Number of rows in the product. Rows within the product indexed row_p = 0, 1, . . . , NumRows – 1.
NumCols Range of values: 1 to 1,000,000.	SICD.ImageData.NumCols	R	INT	Number of columns in the product. Columns within the product indexed col_p = 0, 1, . . . , NumCols – 1.
ICPLat(1)	SICD.Geodata.ImageCorners.ICP.Lat	R	DBL	Image Corner Point 1 latitude (decimal degrees). ICP = 1 ⇔ First row, first column of the product.
ICPLon(1)	SICD.Geodata.ImageCorners.ICP.Lon	R	DBL	Image Corner Point 1 longitude (decimal degrees).
ICPLat(2)	SICD.Geodata.ImageCorners.ICP.Lat	R	DBL	Image Corner Point 2 latitude (decimal degrees).

Table 2-1 SICD Container Placement Parameters

Parameter	XML Field Name	Req / Opt	Type	Description
				ICP = 2 ⇔ First row, last column of the product.
ICPLon(2)	SICD.Geodata.ImageCorners.ICP.Lon	R	DBL	Image Corner Point 2 longitude (decimal degrees).
ICPLat(3)	SICD.Geodata.ImageCorners.ICP.Lat	R	DBL	Image Corner Point 3 latitude (decimal degrees). ICP = 3 ⇔ Last row, last column of the product.
ICPLon(3)	SICD.Geodata.ImageCorners.ICP.Lon	R	DBL	Image Corner Point 3 longitude (decimal degrees).
ICPLat(4)	SICD.Geodata.ImageCorners.ICP.Lat	R	DBL	Image Corner Point 4 latitude (decimal degrees). ICP = 4 ⇔ Last row, first column of the product.
ICPLon(4)	SICD.Geodata.ImageCorners.ICP.Lon	R	DBL	Image Corner Point 4 longitude (decimal degrees).

3 SICD Products In NITF 2.1 Format

This Section provides a detailed definition of how the SICD product is stored in a NITF 2.1 container file.

3.1 SICD in NITF 2.1 Format

The structure of a SICD product in a NITF 2.1 container file is shown in Table 3-1. The NITF 2.1 file consists of a File Header record, one or more Image Segment records and at least one Data Extension Segment record. The data placed in the File Header is defined in Table 3-2 and is based upon Table A-1 of MIL-STD-2500C. The data placed in the Image Sub-Header is defined in Table 3-4 and is based upon Table A-3 of MIL-STD-2500C. The data placed in the Data Extension Segment Sub-Header is defined in Table 3-5 and is based upon Table 1 of the XML_DATA_CONTENT Data Extension Segment Specification.

The mapping of SICD data components to NITF 2.1 file components is shown in Figure 3-1. The example shows a product file with the image pixel array stored in 2 Image Segments. The Container Placement parameters (contained in the XML data) are used to populate fields in the File Header, the Image Sub-Headers and the DES Sub-Header.

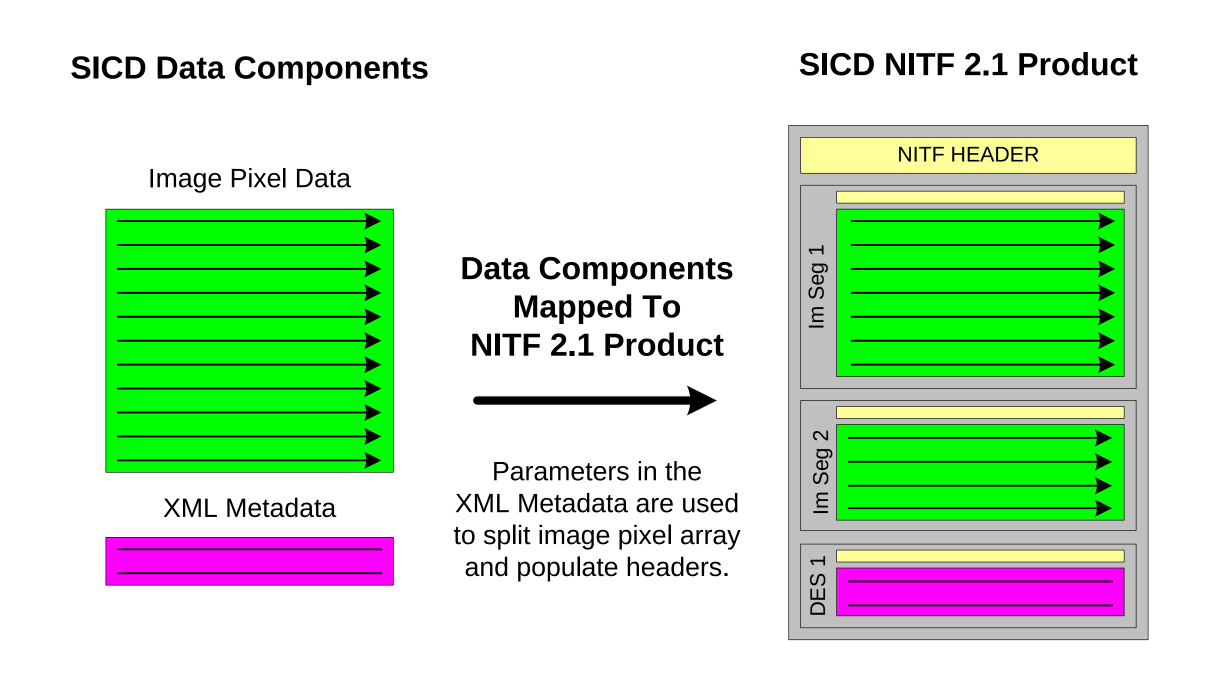

Figure 3-1 SICD NITF 2.1 Product

SICD data components mapped to NITF 2.1 file components.

3.1.1 Format Specifications

The NITF File Header includes fields to identify the product and its source. The File Header includes the number of Image Segments (ISs) contained in the file. The File Header also indicates the number of Data Extension Segment (DES) contained in the file. The SICD image pixel data block is stored in one or more Image Segment(s). The SICD XML Metadata block is stored in the first DES. Security fields are included in the File Header, all Image Sub-Headers and the DES Sub-Header. The security fields are defined in Table 3-3. The NITF encoding is being used to address the Intelligence Community information security marking data elements, which are the essential security markings applicable to the data contained within the file.

The implementation of SICD products for a particular SAR sensor and data processing system includes many program specific design choices. See Section 4.1 of the SICD Design & Implementation Description Document. Program specific implementation guidance is included in what is referred to as a Program Specific Implementation Document. All parameters in Tables 3.2 through 3.5 that require program specific guidance are noted in the tables.

The number of Image Segments used to hold the complex data is determined from the size of the complex pixel data. Each Image Segment can hold up to 9,999,999,998 bytes of data (10^{10} – 2 bytes). If the size of the complex data exceeds the 9,999,999,998 byte limit, it will be placed in multiple Image Segments. If two or more Image Segments (ISs) are used, there is also a maximum number of rows per image segment restriction of 99,999. The number of ISs for the complex data is determined as described in Section 3.2. The complex pixel data is broken across rows when stored in multiple ISs. The next row begins in the next IS so there is continuous flow of data across the segments. The complex data must start in the first Image Segment. The Image Sub-Header fields indicate the portion of the image pixel array stored and its relative placement in the complete image pixel array.

The first Data Extension Segment (DES) contains the SICD XML metadata. The XML_DATA_CONTENT DES is used. The SICD XML metadata is described in the Volume 1, Design & Implementation Description Document. The placement of the XML block in the DES is independent of the precise metadata in the XML block.

The SICD data product does not make explicit use of NITF Tagged Record Extensions (TREs) to describe the format or attributes of the image pixel array. All SICD specific metadata is contained in the SICD metadata block. While the use and inclusion of TREs in the File Header and/or Image Sub-Header is not prohibited, proposed use of TREs with SICD shall be coordinated with the NITFS Technical Board (NTB for authorization, and shall be identified in the Program Specific Implementation Document. Producers of SICD products that choose to include authorized TREs will be responsible for correct formatting and placement. Reader applications developed for SICD products should be designed such that the inclusion NTB-authorized TREs is accommodated.

3.2 Image Segment Sizing Calculations

The computations to determine the number of Image Segments used to contain the image pixel data are described below. The computations also show how to fill NITF fields that vary across Image Segments. Parameter NumIS is the number of Image Segments. Image Segments are indexed n = 1 to NumIS. Parameter NumRowsIS(n) is the number of rows in the nth Image Segment. Parameter FirstRowIS(n) is the product row index (row_p) of the first row in the nth Image Segment. RowOffsetIS(n) is a row offset parameter required for the Image Sub-Header of the nth Image Segment. When the complex pixel array is split across multiple Image Segments the following NITF Image Sub-Header fields are affected: IID1, NROWS/NCOLS, IGEOLO, NPPBH/NPPBV, IDLVL/IALVL and ILOC.

3.2.1 Image Segment Parameters and Equations

SICD Container Placement Parameters – See Table 2-1.

PixelType	Indicates the pixel type and binary format.
NumRows	Rows indexed row_p = 0, 1, 2, . . . , NumRows – 1.
NumCols	Columns indexed col_p = 0, 1, 2, . . . , NumCols – 1.
ICPLat (1), ICPLon(1)	Coordinate for pixel (0,0)
ICPLat (2), ICPLon(2)	Coordinate for pixel (0,NumCols-1)
ICPLat (3), ICPLon(3)	Coordinate for pixel (NumRows-1, NumCols-1)
ICPLat (4), ICPLon(4)	Coordinate for pixel (NumRows-1, 0)

Constants

ISsizeMax = 9,999,999,998

ILOCMax = 99,999

Number of Image Segments & Image Segment Size Parameters

$$\text{BytesPerPixel} = \begin{cases} 8, & \text{if PixelType = RE32F\_IM32F} \\ 4, & \text{if PixelType = RE16I\_IM16I} \\ 2, & \text{if PixelType = AMP8I\_PHS8I} \end{cases}$$

$$\text{BytesPerRow} = \text{BytesPerPixel} \bullet \text{NumCols}$$

$$\text{ProductSize} = \text{BytesPerPixel} \bullet \text{NumRows} \bullet \text{NumCols}$$

$$\text{Limit1} = \text{floor}\left(\frac{\text{ISsizeMax}}{\text{BytesPerRow}}\right) , \qquad \text{Limit2} = \text{ILOCMax}$$

$$\text{NumRowsLimit} = \min(\text{ Limit1, Limit2 })$$

$\textbf{\textit{If}}$ (ProductSize $\leq$ ISsizeMax) $\textbf{\textit{then}}$,

NumIS = 1

NumRowsIS(1) = NumRows

FirstRowIS(1) = 0

RowOffsetIS(1) = 0

Else if (ProductSize > ISsizeMax) ***then,***

$$\text{NumIS} = \text{ceiling}\left(\frac{\text{NumRows}}{\text{NumRowsLimit}}\right)$$

FirstRowIS(1) = 0

RowOffsetIS(1) = 0

For (n = 1,…, NumIS – 1)

NumRowsIS(n) = NumRowsLimit

$\text{FirstRowIS}(n+1) = n \bullet \text{NumRowsLimit}$

RowOffsetIS(n+1) = NumRowsLimit

End

$\text{NumRowsIS(NumIS)} = \text{NumRows} - (\text{NumIS} - 1) \bullet \text{NumRowsLimit}$

End if

Image Segment Corner Coordinate Parameters

The corner coordinates for the portion of the image stored in each Image Segment are computed as follows. For Image Segment n, the corners are indexed 1 to 4 as follows.

ISCPLat(n,1), ISCPLon(n,1)	Coordinate for pixel (FirstRowIS(n),0)
ISCPLat(n,2), ISCPLon(n,2)	Coordinate for pixel (FirstRowIS(n),NumCols-1)
ISCPLat(n,3), ISCPLon(n,3)	Coordinate for pixel (LastRowIS(n),NumCols-1)
ISCPLat(n,4), ISCPLon(n,4)	Coordinate for pixel (LastRowIS(n),0)

where LastRowIS(n) = FirstRowIS(n) + NumRowsIS(n) – 1.

For an image that fits in one Image Segment, assign the corner coordinates of the Image Segment to the corner coordinates of the image pixel array.

ISCPLat(1,1) = ICPLat(1)	ISCPLon(1,1) = ICPLon(1)
ISCPLat(1,2) = ICPLat(2)	ISCPLon(1,2) = ICPLon(2)
ISCPLat(1,3) = ICPLat(3)	ISCPLon(1,3) = ICPLon(3)
ISCPLat(1,4) = ICPLat(4)	ISCPLon(1,4) = ICPLon(4)

For an image that is split across Image Segments, the corner coordinates for each Image Segment are computed as follows. Convert the image pixel array corner positions from latitude and longitude to Earth Centered Fixed (ECF) coordinates. Use a Height Above Ellipsoid = 0.

For each icp = 1, 2, 3, 4:

$$\text{Convert LLH to ECF:} \begin{bmatrix} \text{ICPLat(icp)} \\ \text{ICPLon(icp)} \\ 0 \end{bmatrix} \rightarrow \begin{bmatrix} \text{ICP\_ECFX(icp)} \\ \text{ICP\_ECFY(icp)} \\ \text{ICP\_ECFZ(icp)} \end{bmatrix}$$

For each Image Segment, compute the ECF coordinates for the corners of the first row in the Image Segment (ISCP(n,1) and ISCP(n,2)). Linearly interpolate the ECF positions. Convert the interpolated ECF positions back to LLH.

For n = 1, . . . , NumIS:

$$\text{wgt1(n)} = \frac{\text{NumRows} - 1 - \text{FirstRowIS(n)}}{\text{NumRows} - 1} \qquad \text{wgt2(n)} = \frac{\text{FirstRowIS(n)}}{\text{NumRows} - 1}$$

$$\begin{bmatrix} \text{ISCP\_ECFX(n,1)} \\ \text{ISCP\_ECFY(n,1)} \\ \text{ISCP\_ECFZ(n,1)} \end{bmatrix} = \text{wgt1(n)} \bullet \begin{bmatrix} \text{ICP\_ECFX(1)} \\ \text{ICP\_ECFY(1)} \\ \text{ICP\_ECFZ(1)} \end{bmatrix} + \text{wgt2(n)} \bullet \begin{bmatrix} \text{ICP\_ECFX(4)} \\ \text{ICP\_ECFY(4)} \\ \text{ICP\_ECFZ(4)} \end{bmatrix}$$

$$\text{Convert ECF to LLH:} \quad \begin{bmatrix} \text{ISCP\_ECFX(n,1)} \\ \text{ISCP\_ECFY(n,1)} \\ \text{ISCP\_ECFZ(n,1)} \end{bmatrix} \rightarrow \begin{bmatrix} \text{ISCPLat(n,1)} \\ \text{ISCPLon(n,1)} \\ \text{ISCPHAE(n,1)} \end{bmatrix}$$

$$\begin{bmatrix} \text{ISCP\_ECFX(n,2)} \\ \text{ISCP\_ECFY(n,2)} \\ \text{ISCP\_ECFZ(n,2)} \end{bmatrix} = \text{wgt1(n)} \bullet \begin{bmatrix} \text{ICP\_ECFX(2)} \\ \text{ICP\_ECFY(2)} \\ \text{ICP\_ECFZ(2)} \end{bmatrix} + \text{wgt2(n)} \bullet \begin{bmatrix} \text{ICP\_ECFX(3)} \\ \text{ICP\_ECFY(3)} \\ \text{ICP\_ECFZ(3)} \end{bmatrix}$$

$$\text{Convert ECF to LLH:} \quad \begin{bmatrix} \text{ISCP\_ECFX(n,2)} \\ \text{ISCP\_ECFY(n,2)} \\ \text{ISCP\_ECFZ(n,2)} \end{bmatrix} \rightarrow \begin{bmatrix} \text{ISCPLat(n,2)} \\ \text{ISCPLon(n,2)} \\ \text{ISCPHAE(n,2)} \end{bmatrix}$$

Assign corner coordinates for the last row of each Image Segment as follows. For the first NumIS - 1 image segments, assign the coordinates of the last row to the coordinates of the first row of the next Image Segment. For image segment NumIS, assign the coordinates of the last row to the coordinates of the last row of the image.

For n = 1, . . . , NumIS - 1:

ISCP _Lat(n,3) = ISCP Lat(n+1,2)   ISCP _Lon(n,3) = ISCP Lon(n+1,2)
ISCP _Lat(n,4) = ISCP Lat(n+1,1)   ISCP _Lon(n,4) = ISCP Lon(n+1,1)

For n = NumIS:

ISCP Lat(n,3) = ICPLat(3)   ISCPLon(n,3) = ICPLon(3)
ISCPLat(n,4) = ICPLat(4)   ISCPLon(n,4) = ICPLon(4)

3.2.2 File Header and Image Sub-Header Parameters

Using the parameters computed above, the following File Header and Image Sub-Header parameters can be assigned as described below. The parameter ILOC (RRRRRCCCCC) is filled by concatenating the RowOffsetIS value (RRRRR) and 00000 (CCCCC). The image display level (IDLVL) and the attachment level (IALVL) are set to indicate the connection across Image Segments.

NITF File Header Parameters

File Header NUMI = NumIS

NITF Image Sub-Header Parameters

For n = 1, . . . , NumIS - 1:

Image Sub-Header (n) NROWS = NumRowsIS(n)

Image Sub-Header (n) ILOCRow = RowOffsetIS(n)

Image Sub-Header (n) IDLVL = n

Image Sub-Header (n) IALVL = n – 1

$$\text{Image Sub-Header (n) IGEOLO} = \text{CONCAT}\begin{Bmatrix} \text{ISCPLat(n,1)}, & \text{ISCPLon(n,1)} \\ \text{ISCPLat(n,2)}, & \text{ISCPLon(n,2)} \\ \text{ISCPLat(n,3)}, & \text{ISCPLon(n,3)} \\ \text{ISCPLat(n,4)}, & \text{ISCPLon(n,4)} \end{Bmatrix}$$

Note: See IGEOLO description in Table 3-4.

3.2.3 Example Image Segment Sizing Computations

Example 1. PixelType = RE32F_IM32F NumRows = 2,500 NumCols = 5,000

This image pixel array is placed in one Image Segment.

BytesPerPixel = 8

BytesPerRow = 8 x 5000 = 40,000

ProductSize = 8 x 2,500 x 5,000 = 100,000,000

Limit1 = floor(9,999,999,998 / 40,000) = floor(249,999.999) = 249,999

NumRowsLimit = min(249,999 , 99,999) = 99,999

(ProductSize ≤ ISsizeMax) ➔

NumIS = 1

NumRowsIS(1) = 2,500

FirstRowIS(1) = 0

RowOffsetIS(1) = 0

File Header: NUMI = 1

Image Sub-Header 1: NROWS = 2500, ILOCRow = 0, IDLVL = 1, IALVL = 0

Example 2. PixelType = RE32F_IM32F NumRows = 30,000 NumCols = 90,000

This image pixel array is placed in 3 Image Segments. Image Segments 1 and 2 contain approximately 10**10 bytes each.

BytesPerPixel = 8

BytesPerRow = 8 x 90,000 = 720,000

ProductSize = 8 x 30,000 x 90,000 = 21,600,000,000

Limit1 = floor(9,999,999,998 / 720,000) = floor(13888.888) = 13,888

NumRowsLimit = min(13,888 , 99,999) = 13,888

(ProductSize > ISsizeMax) ➔

NumIS = ceiling(30,000 / 13,888) = ceiling(2.160) = 3

FirstRowIS(1) = 0

RowOffsetIS(1) = 0

For n = 1:2

NumRowsIS(1) = 13,888

FirstRowIS(2) = 1 x 13,888 = 13,888

RowOffsetIS(2) = 13,888

NumRowsIS(2) = 13,888

FirstRowIS(3) = 2 x 13,888 = 27776

RowOffsetIS(3) = 13,888

For n = 3

NumRowsIS(3) = 30,000 – (2 x 13,888) = 2224

File Header: NUMI = 3

Image Sub-Header 1: NROWS = 13,888, ILOCRow = 0, IDLVL = 1, IALVL = 0

Image Sub-Header 2: NROWS = 13,888, ILOCRow = 13,888, IDLVL = 2, IALVL = 1

Image Sub-Header 3: NROWS = 2224, ILOCRow = 13,888, IDLVL = 3, IALVL = 2

Example 3. PixelType = RE16I_IM16I NumRows = 150,000 NumCols = 20,000

This image pixel array is placed in 2 Image Segments. Image Segment 1 contains 99,999 rows (the maximum number of rows per Image Segment when data is split across Image Segments).

BytesPerPixel = 4

BytesPerRow = 4 x 20,000 = 80,000

ProductSize = 4 x 150,000 x 20,000 = 12,000,000,000

Limit1 = floor(9,999,999,998 / 80,000) = floor(124,999.99) = 124,999

NumRowsLimit = min(124,999 , 99,999) = 99,999

(ProductSize > ISsizeMax) ➔

NumIS = ceiling(150,000 / 99,999) = ceiling(1.500) = 2

FirstRowIS(1) = 0

RowOffsetIS(1) = 0

For n = 1:1

NumRowsIS(1) = 99,999

FirstRowIS(2) = 1 x 99,999 = 99,999

RowOffsetIS(2) = 99,999

For n = 2

NumRowsIS(2) = 150,000 – (1 x 99,999) = 50001

File Header: NUMI = 2

Image Sub-Header 1: NROWS = 99,999, ILOCRow = 0, IDLVL = 1, IALVL = 0

Image Sub-Header 2: NROWS = 50001, ILOCRow = 99,999, IDLVL = 2, IALVL = 1

3.3 NITF Header Parameters

Table 3-1 NITF 2.1 File Structure for SICD

NITF File Header	Image (1) Sub-Header	Image (1) Data	. . .	Image (n) Sub-Header	Image (n) Data	DES (1) Sub-Header	DES (1) Data

Table 3-2 NITF 2.1 File Header

Field	Name/Description	Size (bytes)	Value Range	Type	Comment
FHDR	File Profile Name	4	BCS-A NITF	R	Per MIL-STD-2500C
FVER	File Version	5	BCS-A 02.10	R	NITF Version 2.1
CLEVEL	Complexity Level	2	BCS-A 03, 05, 06, 07 or 09	R	Per MIL-STD-2500c Table A-10
STYPE	Standard Type	4	BCS-N BF01	R	Fixed value
OSTAID	Originating Station ID	10	BCS-A This field shall contain a meaningful value; it shall not be filled with BCS spaces (0x20)	R	Per the Program Specific Implementation Document
FDT	File Date and Time	14	BCS-N CCYYMMDDhhmmss	R	File creation UTC date and time
FTITLE	File Title	80	ECS-A "SICD: <corename>"	R	"SICD: <corename>" After SICD colon space fill in with the 1st 74 characters of corename SICD Parameter: CoreName
Security Tags	Use "FS" prefix for Tag	167	As defined in Table 3-3	R	
FSCOP	File Copy Number	5	BCS-N 00000 = No tracking of numbered file copies	R	Not tracked Per MIL-STD-2500c
FSCPYS	File Number of Copies	5	BCS-N 00000 = No tracking of numbered file copies	R	Not tracked Per MIL-STD-2500c
ENCRYP	Encryption	1	BCS-N 0	R	No Encryption
FBKGC	File Background Color	3	Unsigned binary integer 000	R	Default to black (0x00, 0x00, 0x00)
ONAME	Originator's Name	24	ECS-A	<R>	Per the Program

Table 3-2 NITF 2.1 File Header

Field	Name/Description	Size (bytes)	Value Range	Type	Comment
			24 ECS characters		Specific Implementation Document
OPHONE	Originator's Phone	18	ECS-A 24 ECS characters	<R>	Per the Program Specific Implementation Document
FL	File Length	12	BCS-N Generate	R	Number of bytes
HL	NITF File Header Length	6	BCS-N Generate	R	Number of bytes in the header
NUMI	Number of Image Segments	3	BCS-N 001 to 999	R	Number of Image Segments included in the file. At least the number computed for the image data **See Section 3.2**
. . . . . Start for each IS LISHn, LIn.					
NOTE: LISHn and LIn fields repeat in pairs such that LISH001, LI01; LISH002, LI002; .....LISHn,LIn.					
LISHn	Length of Image Sub-Header. This field shall occur as many times as specified in the NUMI field. Note: The largest image sub-header is limited to 999998 (10**6 -2) bytes.	6	BCS-N 000439 to 9999998 Generate	C	Per MIL-STD-2500c
LIn	Length of nth Image Segment. This field shall occur as many times as specified in the NUMI field. Note: The largest image is limited to 9,999,999,998 (10**10 -2) bytes.	10	BCS-N 0000000001 to 9999999998 Generate	C	Number of Image data bytes in Image Segment n. NumRowsIS(n) x NumCols x BytesPerPixel **See Section 3.2**
. . . . End for each IS LISHn, LIn; the number of loop repetitions is the value specified in the NUMI field.					
NUMS	Number of graphic Segments	3	BCS-N 000	R	Not used default to 000
NUMX	Reserved	3	BCS-N 000	R	Reserved
NUMT	Number of Text Segments	3	BCS-N 000 or 001 to 999	R	If Text Segments Not Used = 000 If used, value = number of Text Segments
LTSHn and LTn repeat as pairs as many times as specified in the NUMT field					

Table 3-2 NITF 2.1 File Header					
Field	**Name/Description**	**Size (bytes)**	**Value Range**	**Type**	**Comment**
LTSHn	Length of nth Text Segment Sub-Header	4	BCS-N 0282 to 9998	C	Length in bytes
LTn	Length of Text Segment	5	00001 to 99998	C	Length in bytes
. . . . End for each IS LTSHn, LTn; the number of loop repetitions is the value specified in the NUMT field.					
NUMDES	Number of data Extension Segments	3	BCS-N 001 to 999	R	Number of separate DES included in the file. NUMDES = 001 when only including XML.
LDSHn and LDn repeat as pairs as many times as specified in the NUMDES field					
LDSHn	Length of nth Data Extension Segment Sub-Header	4	BCS-N 0200 to 9998	R	Length in bytes
LD1n	Length of Data Extension Segment	9	BCS-N 000000001 to 999999998	R	Length in bytes
End of LDSH and LDn field repetition					
NUMRES	Number of Reserved	3	BCS-N 000	R	Not Used
UDHDL	User-Defined Header Data Length	5	BCS-N 00000 00003 + length of TREs	R	No TREs specific to SICD products but use of TREs is not prohibited.
UDHOF	User-Defined Header Overflow	3	BCS-N 000	C	If UDHDL = 00000 then omit field. No overflow otherwise.
UDHD	User-Defined Header Data	3	BCS-N Equal to UDHL - 3	C	If UDHDL = 00000 then omit field.
XHDL	Extended Header Data Length	5	BCS-N 00000 00003 + length of TREs	R	No TREs specific to SICD products but use of TREs is not prohibited.
XHD	Extended Header Data	5	BCS-N Equal to XHDL - 3	C	If XHDL = 00000 then omit field.
Type – R = Required, C = Conditional, <> surrounding the type = BCS SPACES ALLOWED FOR ENTIRE FIELD					

Table 3-3 NITF 2.1 Security Tags				
Base Field	**Name/Description**	**Size (bytes)**	**Value Range**	**Type**
xxCLAS	File Security Classification. 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).	1	ECS-A "U" or Per the Program Specific Implementation Document	R
xxCLSY	File Security Classification System. This field shall contain valid values indicating the national or multinational security system used to classify the file. Country Codes per FIPS PUB 10-4 shall be used to indicate national security systems. The designator "XN" is for classified data generated by a component using NATO security system marking guidance. This code is outside the FIPS 10-4 document listing, and was selected to not duplicate that document's existing codes. If this field is all ECS spaces (0x20), it shall imply that no security classification system applies to the file.	2	ECS-A (Default is ECS spaces (0x20)) Per the Program Specific Implementation Document	<R>
xxCODE	File Codewords. This field shall contain a valid indicator of the security compartments associated with the file. Values include one or more of the digraphs found in MIL-STD-2500c, table A-4. Multiple entries shall be separated by a single ECS space (0x20): The selection of a relevant set of codewords is application specific. If this field is all ECS spaces (0x20), it shall imply that no codewords apply to the file.	11	BCS-A (Default is BCS spaces (0x20)) Per the Program Specific Implementation Document	<R>
xxCTLH	File Control and Handling. This field shall contain valid additional security control and/or handling instructions (caveats) associated with the file. Values include digraphs found in MIL-STD-2500c, table A-4. The digraph may indicate single or multiple caveats. The selection of a relevant caveat(s) is application specific. If this field is all ECS spaces (0x20), it shall imply that	2	ECS-A (Default is ECS spaces (0x20)) Per the Program Specific Implementation Document	<R>

Table 3-3 NITF 2.1 Security Tags				
Base Field	**Name/Description**	**Size (bytes)**	**Value Range**	**Type**
	no additional control and handling instructions apply to the file.			
xxREL	File Releasing Instructions. This field shall contain a valid list of country and/or multilateral entity codes to which countries and/or multilateral entities the file is authorized for release. Valid items in the list are one or more country codes as found in FIPS PUB 10-4 separated by a single ECS space (0x20). If this field is all ECS spaces (0x20), it shall imply that no file release instructions apply.	20	ECS-A (Default is ECS spaces (0x20)) Per the Program Specific Implementation Document	<R>
xxDCTP	File Declassification Type. This field shall contain a valid indicator of the type of security declassification or downgrading instructions which apply to the file. Valid values are DD (=declassify on a specific date), DE (=declassify upon occurrence of an event), GD (=downgrade to a specified level on a specific date), GE (=downgrade to a specified level upon occurrence of an event), O (=OADR), and X (= exempt from automatic declassification). If this field is all ECS spaces (0x20), it shall imply that no file security declassification or downgrading instructions apply.	2	ECS-A (Default is ECS spaces (0x20)) Valid Values DD, DE, GD, GE, O, X Per the Program Specific Implementation Document	<R>
xxDCDT	File Declassification Date. This field shall indicate the date on which a file is to be declassified if the value in File Declassification Type is DD. If this field is all ECS spaces (0x20), it shall imply that no file declassification date applies.	8	ECS-A (Default is ECS spaces (0x20)) Valid Format CCYYMMDD Per the Program Specific Implementation Document	<R>
xxDCXM	File Declassification Exemption. This field shall indicate the reason the file is exempt from automatic declassification if the value in File Declassification Type is X. Valid values are X1 to X8 and X251 to X259. X1 to X8 correspond to the declassification exemptions found in DOD 5200.1-R, paragraphs 4- 202b(1) to (8) for material exempt from the 10-year rule. X251 to X259 correspond to	4	ECS-A (Default is ECS spaces (0x20)) Valid values X1 to X8, X251 to X259 Per the Program Specific Implementation Document	<R>

Table 3-3 NITF 2.1 Security Tags

Base Field	Name/Description	Size (bytes)	Value Range	Type
	the declassification exemptions found in DOD 5200.1-R, paragraphs 4- 301a(1) to (9) for permanently valuable material exempt from the 25-year declassification system. If this field is all ESC spaces (0x20), it shall imply that a file declassification exemption does not apply.			
xxDG	File Downgrade. This field shall indicate the classification level to which a file is to be downgraded if the values in File Declassification Type are GD or GE. Valid values are S (=Secret), C (=Confidential), R (= Restricted). If this field contains an ECS space (0x20), it shall imply that file security downgrading does not apply.	1	ECS-A (Default is ECS space (0x20)) Valid Values S, C, R Per the Program Specific Implementation Document	<R>
xxDGDT	File Downgrade Date. This field shall indicate the date on which a file is to be downgraded if the value in File Declassification Type is GD. If this field is all ECS spaces (0x20), it shall imply that a file security downgrading date does not apply.	8	ECS-A (Default is ECS spaces (0x20)) Valid Format CCYYMMDD Per the Program Specific Implementation Document	<R>
xxCLTX	File Classification Text. This field shall be used to provide additional information about file classification to include identification of a declassification or downgrading event if the values in File Declassification Type are DE or GE. It may also be used to identify multiple classification sources and/or any other special handling rules. Values are user defined free text. If this field is all ECS spaces (0x20), it shall imply that additional information about file classification does not apply.	43	ECS-A (Default is ECS spaces (0x20)) Per the Program Specific Implementation Document	<R>
xxCATP	File Classification Authority Type. This field shall indicate the type of authority used to classify the file. Valid values are O (= original classification authority), D (= derivative from a single source), and M (= derivative from multiple sources). If this field contains an ECS space (0x20), it shall imply that file classification	1	ECS-A (Default is ECS space (0x20)) Per the Program Specific Implementation Document	<R>

Table 3-3 NITF 2.1 Security Tags

Base Field	Name/Description	Size (bytes)	Value Range	Type
	authority type does not apply.			
xxCAUT	File Classification Authority. This field shall identify the classification authority for the file dependent upon the value in File Classification Authority Type. Values are user defined free text which should contain the following information: original classification authority name and position or personal identifier if the value in File Classification Authority Type is O; title of the document or security classification guide used to classify the file if the value in File Classification Authority Type is D; and Derive-Multiple if the file classification was derived from multiple sources and the value of the FSCATP field is M. In the latter case, the file originator will maintain a record of the sources used in accordance with existing security directives. One of the multiple sources may also be identified in File Classification Text if desired. If this field is all ECS spaces (0x20), it shall imply that no file classification authority applies.	40	ECS-A (Default is ECS spaces (0x20)) Per the Program Specific Implementation Document	<R>
xxCRSN	File Classification Reason. This field shall contain values indicating the reason for classifying the file. Valid values are A to G. These correspond to the reasons for original classification per E.O. 12958, Section 1.5.(a) to (g). If this field contains an ECS space (0x20), it shall imply that no file classification reason applies.	1	ECS-A (Default is ECS space (0x20)) Valid Values A to G Per the Program Specific Implementation Document	<R>
xxSRDT	File Security Source Date. This field shall indicate the date of the source used to derive the classification of the file. In the case of multiple sources, the date of the most recent source shall be used. If this field is all ECS spaces (0x20), it shall imply that a file security source date does not apply.	8	ECS-A (Default is ECS spaces (0x20)) Valid Format CCYYMMDD Per the Program Specific Implementation Document	<R>
xxCTLN	File Security Control Number. This field shall contain a valid security control number associated with the file. The	15	ECS-A (Default is ECS spaces (0x20)) Per the Program Specific	<R>

Table 3-3 NITF 2.1 Security Tags

Base Field	Name/Description	Size (bytes)	Value Range	Type
	format of the security control number shall be in accordance with the regulations governing the appropriate security channel(s). If this field is all ECS spaces (0x20), it shall imply that no file security control number applies.		Implementation Document	
For the File Header replace the xx prefix with FS (e.g., FSCLAS) For the Image Sub-Header replace the xx prefix with IS (e.g., ISCLAS) For the DES replace the xx prefix with DES (e.g., DESCLAS)				
Refer to MIL-STD-2500C Table A-1 for File Header and Table A-3 for Image Sub-Header for general description of all fields. Consult current security guidelines at time of production for proper entries.				
Type – R = REQUIRED, C = CONDITIONAL, <> surrounding the type = BCS SPACES ALLOWED FOR ENTIRE FIELD				

Table 3-4 NITF 2.1 Image Sub-Header

Field	Name/Description	Size (bytes)	Value Range	Type	Comment
IM	File Part Type.	2	BCS-A IM	R	
IID1	Image Identifier 1.	10	BCS-A SICDnnn nnn has a range 000, 001-999	R	For complex images contained in a single IM segment, nnn = 000 For complex images contained in two or more IM segments, populate nnn with 001 for the first segment and progress sequentially for each segment used to contain the image.
IDATIM	Image Date and Time.	14	BCS-N CCYYMMDDhhmmss	R	Collection Start Time in UTC SICD Parameter: CollectStart
TGTID	Target Identifier.	17	BCS-A BBBBBBBBBB_____CC (Default is BCS spaces (0x20) for all or any sub-part of this field) Characters 11–15 BCS spaces	<R>	If known, the first 10 characters may contain the BE Identifier. If known, the last 2 characters (16 - 17) may contain the Country Code This is intended for general reference. Other Target Identifiers defined in XML data
IID2	Image Identifier 2.	80	ECS-A SICD: Default to value used for FTITLE. Program Specific guidance may override.	R	Per the Program Specific Implementation Document.
Security Tags	Use “IS” prefix for Tags	167	As defined in Table 3-3	R	
ENCRYPT	Encryption.	1	BCS-N positive integer 0	R	0 = not encrypted
ISORCE	Image Source.	42	ECS-A SICD: Fill with meaningful value.	R	Per the Program Specific Implementation Document
NROWS	Number of Significant Rows in Image.	8	BCS-N positive integer 00000001 to 99999999	R	Number of significant rows in image segment NumRowsIS(n)

Table 3-4 NITF 2.1 Image Sub-Header

Field	Name/Description	Size (bytes)	Value Range	Type	Comment
					See Section 3.2
NCOLS	Number of Significant Columns in Image.	8	BCS-N positive integer 00000001 to 99999999	R	Number of significant columns in image segment SICD Parameter: NumCols **See Section 3.2**
PVTYPE	Pixel Value Type	3	BCS-A "R" followed by a space, "SI" followed by a space, or "INT"	R	R for Real SI for 2's complement signed integer INT for unsigned integer SICD Parameter: PixelType
IREP	Image Representation.	8	BCS-A NODISPLY	R	SICD is complex image data that is not display ready without further processing
ICAT	Image Category.	8	BCS-A SAR	R	"SAR" for Synthetic Aperture Radar
ABPP	Actual Bits-Per-Pixel Per Band.	2	BCS-N positive integer 08, 16 or 32	R	SICD Parameter: PixelType RE32F_IM32F = 32 RE16I_IM16I = 16 AMP8I_PHS8I = 08 **See Section 2.3**
PJUST	Pixel Justification.	1	BCS-A R	R	Default is R Field needed if ABPP and NPBB are not equal. This is not the case for SICD
ICORDS	Image Coordinate Representation.	1	BCS-A G	R	G for geographic
IGEOLO	Image Geographic Location.	60	BCS-A ddmmssXdddmmssY (four times)	C	Calculate from complex image corner coordinates. **See Section 3.2** ddd = degree mm = minute ss = seconds X = "N" or "S" Y = "E" or "W"

Table 3-4 NITF 2.1 Image Sub-Header

Field	Name/Description	Size (bytes)	Value Range	Type	Comment
NICOM	Number of Image Comments.	1	BCS-N positive integer 0 to 9	R	
NICOMn	Image Comment n.	80	ECS-A User defined	C	
IC	Image Compression.	2	BCS-A NC	R	“NC” for No Compression
NBANDS	Number of Bands.	1	BCS-N positive integer 2	R	2 bands for the Real & Imaginary or Magnitude & Phase image data
IREPBANDn	nth Band Representation.	2	BCS-A BCS spaces (0x20)	<R>	Default value, data not intended for display
ISUBCATn	nth Band Subcategory.	6	BCS-A “I”, “Q” or “M”,”P” followed by spaces For SICD: RE ➔ Inphase ➔”I” IM ➔ Quadrature ➔”Q” AMP ➔ Magnitude ➔”M” PHS ➔ Phase ➔”P”	R	“I” ⇔ Inphase band “Q” ⇔ Quadrature band “M” ⇔ Magnitude band “P” ⇔ Phase band SICD Parameter: PixelType
IFCn	nth Band Image Filter Condition.	1	BCS-A N	R	
IMFLTn	nth Band Standard Image Filter Code.	3	BCS-A Fill with BCS spaces (0x20)	<R>	3 spaces
NLUTSn	Number of LUTS for the nth Image Band.	1	BCS-N positive integer 0	R	0 = No Look Up Tables
ISYNC	Image Sync code.	1	BCS-N positive integer 0 = No Sync Code	R	0 = No sync code
IMODE	Image Mode.	1	BCS-A P represents Band Interleaved by Pixel.	R	“P” for pixel interleaved
NBPR	Number of Blocks Per Row.	4	BCS-N positive integer 0001	R	
NBPC	Number of Blocks Per Column.	4	BCS-N positive integer 0001	R	
NPPBH	Number of Pixels Per Block Horizontal.	4	BCS-N positive integer 0000 or 0001 to 8192	R	if NCOLS > 8192 value = 0000 if NCOLS <= 8192 value = NCOLS
NPPBV	Number of Pixels Per Block Vertical.	4	BCS-N positive integer 0000 or 0001 to 8192	R	if NROWS > 8192 value = 0000 if NROWS <= 8192

Table 3-4 NITF 2.1 Image Sub-Header					
Field	**Name/Description**	**Size (bytes)**	**Value Range**	**Type**	**Comment**
					value = NROWS
NBPP	Number of Bits Per Pixel Per Band.	2	BCS-N positive integer 08, 16 or 32	R	SICD Parameter PixelType RE32F_IM32F = 32 RE16I_IM16I = 16 AMP8I_PHS8I = 8 **See Section 2.3**
IDLVL	Image Display Level.	3	BCS-N positive integer 001 to 999	R	Increase by image segment, start at 001 **See Section 3.2**
IALVL	Attachment Level.	3	BCS-N positive integer 000 to 998	R	Increase by image segment, start at 000 Value corresponds to display level -1. If multiple segments are needed, the attachment level should always match the previous image segments display level. **See Section 3.2**
ILOC	Image Location.	10	BCS-N RRRRR00000	R	RRRRRCCCCC R is for Row C is for Column The first segment RRRRR = 00000. If multiple segments are needed the RRRRR value should match the NROWS in the previous segment. **See Section 3.2**
IMAG	Image Magnification.	4	BCS-A 1.0 followed by BCS space (0x20)	R	1.0
UDIDL	User Defined Image Data Length.	5	BCS-N positive integer 00000	R	Generally not used for SICD Image Segments. See 3.1.1
IXSHDL	Image Extended Sub-Header Data Length.	5	BCS-N positive integer 00000	R	Generally not used for SICD Image Segments. See 3.1.1
Type – R = Required, C = Conditional, <> surrounding the type = BCS SPACES ALLOWED FOR ENTIRE FIELD					

Table 3-5 NITF 2.1 XML_DATA_CONTENT Data Extension Segment Sub-Header					
Field	**Name/Description**	**Size**	**Value Range**	**Type**	**Comment**
DE	File Part Type	2	BCS-A DE	R	Identifies as a DES Sub-Header
DESID	Unique DES Type Identifier. This field shall contain a valid alphanumeric identifier properly registered with the ISMC.	25	BCS-A "XML_DATA_CONTENT" followed by 10 spaces	R	Identifies the DES
DESVER	Version of the Data Definition. This field shall contain the alphanumeric version number of the use of the tag. The version number is assigned as part of the registration process.	2	BCS-N 01	R	
Security Tags	See XML_DATA_CONTENT Data Extension Segment Specification	167	As defined in Table 3-3	R	
DESSHL	DES User-defined Subheader Length. This field shall contain the number of bytes in the field DESSHF. The inclusion of DESSHF subfield content is conditional based on the byte count value entered in this field. The DESSHF field may be omitted by setting this value to 0000. Otherwise, three increments of subfield inclusion are identified in the specified value range.	4	BCS-N positive integer 0773 – Complete inclusion of all User-defined Subheader Subfields.	R	SICD: All user-defined sub-header fields are included.
Note: The 'conditional' field, DESSHF (DES User-defined Subheader Field) is included in all SICD products. The DESSHF is comprised of the sub-fields DESCRC through DESSHABS specified below.					
DESCRC	Cyclic Redundancy Check. This field contains the calculated CRC value for the content of the DESDATA field. A value of 99999 shall be used when CRC is not calculated.	5	BCS-N positive integer 99999	C	CRC is not computed for SICD products.
DESSHFT	XML File Type. Data in this field shall be representative of the XML File Type. Examples: XSD, XML, DTD, XSL, XSLT	8	BCS-A XML	C	Indicates the data contained the DES is the XML instance.
DESSHDT	Date and Time. This field shall contain the time (UTC) of the XML file's origination in the format: YYYY-MM-DDThh:mm:ssZ, where YYYY is the year (0000-9999), MM is the month (01 to 12), DD is the day (01 to 31), T is the separator between date and time, hh is the hour (00 to 23), mm is the minute (00 to 59), and ss is the second (00 to 59).	20	BCS-A YYYY-MM-DDThh:mm:ssZ	C	
DESSHRP	Responsible Party – Organization Identifier. Identification of the organization responsible for the content of the	40	U8 Free Text Per Program Specific	C	

	DES.		Implementation Document		
DESSHSI	Specification Identifier. Name of the specification used for the XML data content.	60	U8 SICD Volume 1 Design & Implementation Description Document	C	
DESSHSV	Specification Version. Version or edition of the specification.	10	BCS-A Free Text	C	
DESSHSD	Specification Date. Version or edition date for the specification. See Date and Time description above.	20	BCS-A YYYY-MM-DDThh:mm:ssZ	C	
DESSHTN	Target Namespace. Identification of the target namespace, if any, designated within the XML data content.	120	BCS-A URI/URN/URL Default is BCS spaces (0x20)	<C>	Specify the SICD URN.
DESSHLPG	Location – Polygon. Five-point boundary enclosing the area applicable to the DES, expressed as the closed set of coordinates of the polygon (last point replicates first point). NOTE: This is only an approximate reference so specifying the coordinate reference system is unnecessary. Recorded as paired latitude and longitude values in decimal degrees with no separator. Each latitude and longitude value includes an explicit 'plus' or 'minus sign'. The precision for recording the values in the subheader is dictated by the field size constraint.	125	BCS-A Five pairs of latitude and longitude values. -90 to +90 latitude -180 to +360 longitude ±dd.dddddddd±ddd.ddddddddd ±dd.dddddddd±ddd.ddddddddd ±dd.dddddddd±ddd.ddddddddd ±dd.dddddddd±ddd.ddddddddd ±dd.dddddddd±ddd.ddddddddd	<C>	Use image corner point locations. See Table 2-1. Order is: ICP1, ICP2, ICP3, ICP4, ICP1.
DESSHLPT	Location – Point. Single geographic point applicable to the DES. NOTE: This is only an approximate reference so specifying the coordinate reference system is unnecessary. Recorded as paired latitude and longitude values in decimal degrees. The precision for recording the values is dictated by the user application and the field size constraint. The latitude and longitude value includes an explicit 'plus' or 'minus'.	25	BCS-A BCS spaces (0x20) Default is BCS spaces (0x20)	<C>	Fill with spaces.
DESSHLI	Location – Identifier. Identifier used to represent a geographic area. An alphanumeric value identifying an instance in the designated namespace. When this	20	BCS-A Per Program Specific Implementation Document	<C>	

	field is recorded with other than the default value, the Location Identifier Namespace URI shall also be recorded. Examples: US USA		Default is BCS spaces (0x20)		
DESSHLIN	Location Identifier Namespace URI. URI for the Namespace where the Location Identifier is described. Example: http://metadata.dod.mil/mdr/ns/GPAS/codelist/ fips10-4/digraph http://metadata.dod.mil/mdr/ns/GPAS/codelist/ iso3166-1/trigraph	120	BCS-A Default is BCS spaces (0x20)	<C>	
DESSHABS	Abstract. Brief narrative summary of the content of the DES.	200	U8 Free Text Per Program Specific Implementation Document Default is BCS spaces (0x20)	<C>	
DESDATA	DES User-defined Data This field shall contain the XML data. The length of this field shall not cause any other NSIF/NITF field length limits to be exceeded.	Note 1	Total length DESDATA cannot exceed 999999998 bytes.	R	
Note 1: Determined by user.					
Type – R = REQUIRED, C = CONDITIONAL, <> surrounding the type = DESIGNATED DEFAULT VALUE ALLOWED FOR ENTIRE FIELD					

4 Appendix A

Terms and Definitions	
Term	**Definition**
Attachment Level (ALVL)	A way to associate images and graphics to the same level during movement, rotation, or display.
Base Image	A base image is the principle image of interest or focus for which other data may be inset or overlaid. The NITF file can have none, one, or multiple base images. For multiple base images in a single NITF file, the relative location of each base image is defined in the image location (ILOC) field in each image sub-header. This location will be the offset within the Common Coordinate System (CCS).
Basic Character Set (BCS)	A subset of the Extended Character Set (ECS). The most significant bit of the BCS characters is set to 0. The range of valid BCS characters code is limited to 0x20 to 0x7E plus line feed (0x0A), form feed (0x0C), and carriage return, (0x0D).
Basic Character Set-Alphanumeric (BCS-A)	A subset of the Basic Character Set. The range of allowable characters consists of space to tilde, codes 0x20 to 0x7E.
Basic Character Set-Numeric Integer (BCS-N integer)	A subset of the Basic Character Set- Numeric (BCS-N) comprising the digits 0 to 9 (codes 0x30 to 0x39), plus sign (code 0x2B) and minus sign (code 0x2D).
Basic Character Set – Numeric Positive Integer (BCS-N positive integer)	A subset of the Basic Character Set Numeric (BCS-N) comprising the digits 0 to 9 (codes 0x30 to 0x39). MIL-STD-2500C.
BCS Space	BCS (and consequently ECS) code 0x20.
Common Coordinate System (CCS)	The virtual two dimensional Cartesian-like coordinate space which will be common for determining the placement and orientation of displayable data.
Complexity Level (CLEVEL)	A code used in the file header which signals the degree of complexity an interpret implementation needs to support to adequately interpret the files. Items that differentiate complexity include: size of the common coordinate system, file size, image size, image blocking, color of imagery with greater than 8-bit per pixel, number of bands in an image segment, number of image segments, aggregate size of graphic segments, etc..
Coordinated Universal Time (UTC)	The time scale maintained by the International Earth Rotation Service (having previously been maintained by the Bureau International de l'Heure that forms the basis of a coordinated dissemination of standard frequencies and time signals.
Data Extension Segment (DES)	A type of extension segment with sub-header and data fields structured similarly to the standard data types in the NITF (e.g., image, label, symbol, text). The extension type identifier (25 character DESTAG field), the version (two character DESVER field), and the full underlying structure is under configuration management control as registered with the NTB.
Extended Character Set (ECS)	A set of 1-byte encoded characters. Valid ECS character codes range from 0x20 to 0x7E, and 0xA0 to 0xFF, as well as Line Feed (0x0A), Form Feed (0x0C) and Carriage Return (0x0D). The ECS characters are described in Table B-1. As an interim measure, because of inconsistencies between standards, it is strongly advised that character codes ranging from 0xA0 to 0xFF should never be used. Therefore, the use of ECS characters should be restricted to its BCS Subset.
Extended Character Set (ECS) Space	See BCS Space definition.

Terms and Definitions	
Term	**Definition**
Extended Character Set - Alphanumeric (ECS-A)	A subset of the Extended Character Set (ECS). Valid ECS-A character codes range from 0x20 to 0x7E, and 0xA0 to 0xFF. Line Feed (0x0A), Form Feed (0x0C) and Carriage Return (0x0D) are not valid ECS-A characters. As an interim measure, because of inconsistencies between standards, it is strongly advised that character codes ranging from 0xA0 to 0xFF should never be used. Therefore, the use of ECS-A characters should be restricted to its BCS-A Subset.
Non-blank	Non-blank indicates that the field cannot be filled by the character space (0x20) but may contain the character space when included with other characters.. (embedded blanks).
Null	The field is filled entirely with spaces (0x20).
Segment	A header and data fields.