Astron. Astrophys. Suppl. Ser. 44, (1981) 363-370

FITS : A FLEXIBLE IMAGE TRANSPORT SYSTEM

D. C. WELLS (1), E. W. GREISEN (*) (2) and R. H. HARTEN (**)

Kitt Peak National Observatory Tucson, Arizona, U.S.A.
(*) National Radio Astronomy Observatory Charlottesville, Virginia, U.S.A.
(**) The Netherlands Foundation for Radio Astronomy, Postbus 2, 7990 AA Dwingeloo, the Netherlands

Received March 31, accepted September 4, 1980

Summary.- A format for the interchange of astronomical images and other digital arrays on magnetic tape is described. This format provides a simple but powerful mechanism for the unambiguous transmission of n-dimensional, regularly spaced data arrays. It also provides a method for the transmission of a virtually unlimited number of auxiliary parameters that may be associated with the image. The parameters are written in a form which is easily interpreted by both humans and computers. The FITS format has been adopted for the transmission of astronomical image data by several large observatories including the Very Large Array, the Westerbork synthesis telescope, the Kitt Peak Observatory and the Anglo-Australian Observatory.

Key words : data analysis

1. Introduction.- With the advent of the WSRT and the VLA in radio astronomy, the increased use of CCD arrays and other digital techniques in optical astronomy, and the development of satellites for astronomical observations at other frequencies, the number of images in digital form has increased enormously. In order to compare images made at different observatories and in order to continue the processing of the images, astronomers often wish to transport their data to their home sites or to other observatories. This interchange of data has traditionally been hampered by the fact that each installation has generated its own software system for image processing tailored to its own computer facilities, which differ enormously. Almost every installation has developed at least one unique data format and produced a large quantity of software based on the use of that internal format. Given this situation, the adoption of a single format for use in all installations would be prohibitively expensive and would lead, in general, to less efficient computing within the individual systems. However, a feasible course of action is the adoption of a unique interchange tape format to be used for transferring digital imagery between cooperating institutions. Each institution wishing to exchange imagery would then need to write only two programs - one to translate the transfer format into the internal format used by that institution and one to perform the reverse translation.

A unique image interchange format needs to be very flexible. It must facilitate the unambiguous transmission of n-dimensional, regularly spaced data arrays. It should provide a mechanism for transmitting any auxiliary parameters that are associated with the image, even though not all of these parameters, or even the nature of all of these parameters, can be specified a priori. It should provide the means to transmit arbitrary amounts of text within standard data files in order to encourage the use of "self-documenting" data tapes. Finally, it needs to be general so that it can adapt to new situations and other inevitable changes, and so that it can be used in image processing applications outside of astronomy.

Over the past several years we have developed and used a number of formats in an attempt to find a solution to the data transport problem. These experiences confirmed the validity of the basic idea, but pointed out the dangers of rigidity and complexity in the design of interchange formats. We have reviewed our formats and a variety of other existing formats (e.g. "Workshop on Standards for Image Pattern Recognition", 1977) and concluded that none of them contained adequate simplicity, flexibility or generality. Accordingly, we designed an entirely new interchange format. Our scheme is described in an expanded, somewhat tutorial fashion in the following text and is summarized in Appendix A. We feel that this FITS format meets the criteria given above. It provides for the unambiguous transfer of images that have accuracies up to 32 bits, including sign, with a bare minimum of required header information. The exporting institution may choose to write any amount of secondary header information in a form that is readily interpretable by both humans and computers. The receiving institution may write a simple translation program that will recover most images with very little difficulty, or it may write a more advanced program that will decode the more complicated image types and ancillary header data allowed by the format.

As used in this paper, an "image" is any n-dimensional, regularly spaced array of data and a "pixel" (or picture element) is any single location within that array. Much of the data produced by modern ground-based and satellite instruments is in the form of images having one, two, three, or even more dimensions. This is true for radio, infrared, optical, and X-ray data. Typically the image represents the source intensity at a grid of points in one or two spatial coordinates, time, frequency, and/or polarization. However,

(1) Operated by the Association of Universities for Research in Astronomy, Inc., under contract with the National Science Foundation.

(2) Operated by Associated Universities, Inc., under contract with the National Science Foundation.

it may represent less direct quantities such as spectral index, velocity, and polarization position angle on a grid in time and/or spatial coordinates. FITS provides a convenient means to transport all of these data types and many more. At this time, FITS has been adopted and, at least partially, implemented by the National Radio Astronomy Observatory (for map data from the Very Large Array), by the Netherlands Foundation for Radio Astronomy (for map data from the Westerbork Synthesis Radio Telescope), and by the Kitt Peak National Observatory (for optical data). Several other major astronomical institutions have already expressed interest in, and support for, this system.

2. The physical representation of the data.- FITS is a data format for recording digital images on seven-track and nine-track magnetic tape. The arrangement of the digital bits and the other physical properties of such tapes shall conform to the relevant ANSI (American National Standards Institute) specifications. For simplicity, each physical record on the tape will contain one and only one logical record. Furthermore, all records will have a length of 23040 bits (2880 8-bit bytes, 3840 6-bit bytes). This length is evenly divisible by both the byte and word lengths of all computers that have been sold in the commercial market (i.e., 6, 8, 12, 16, 18, 24, 32, 36, 48, 60, and 64 bits). We believe that this universal commensurability facilitates packing and unpacking the records on a wide variety of computers. The record length is long enough to provide adequate packing efficiency for all standard types of magnetic tape (e.g., 83 % for 800 bpi nine-track tape, 75 % for 1600 bpi, and 61 % for 6250 bpi). However, it is not so long as to pose a problem for small computers.

One tape file will be used for each image. Each file will begin with one or more logical records containing "header" data in the form of 36 80-character (8-bits per character) card images per record. The character code will be 7-bit ASCII (also known as ISO/R646 and CCITT Alphabet # 5), right justified in the 8-bit bytes. The high-order bit of each byte (the "parity" bit) will be zero. The contents of the header card images are discussed in sections 3, 4 and 5.

The first data array record will occur immediately after the last header record. The first image pixel value will occur in the first pixel position in this first data array record. The first pixel of each row will always be put into the next available pixel position in the current logical record. Thus the image data will be packed into records with maximum efficiency, regardless of the commensurability of the row length (which can be any positive integer) with the record length (2880 bytes). The remainder of the last data array record after the last pixel of the array will be padded with zeros. We will support three types of pixel value representations : 8-bit unsigned binary integers, 16-bit twos-complement signed binary integers, and 32-bit twos-complement signed binary integers. We require that the 2 or 4 bytes of 16 or 32-bit pixel values be written on the tape in order of decreasing significance. The byte that includes the sign bit will be first, and the byte that has the 1-bit of the 16 or 32-bit integer as its least significant bit will be last. Thus users of the DEC PDP-11 series will be obliged to correct for the "byte-swap" action of that series' architecture.

The file of data records representing the image is to be terminated by a tapemark. A program which is to read the file should be prepared to skip any records that may occur after the last data array record and before the tapemark. This convention permits the invention of new record types in the future. We merely require that they must come after the data array records, and that they have a length of 23040 bits. The conventions for our header parameters allow the dimensions of the FITS data array to be zero. Thus the possibility exists that the normal data array may not be present, and that records of some new type may appear instead. Any number of files may be written and there is no requirement that all of the files on the tape have the same dimensions. At least one extra tapemark should be written after the tapemark which terminates the last data file on the tape.

The FITS standard allows both ANSI labelled tapes and unlabelled tapes, but we encourage the use of single-volume unlabelled tapes. A recipient of a labelled tape may, of course, treat it as unlabelled by processing tape files 2, 5, 8, 11, ... as image files 1, 2, 3, 4, ... The use of ANSI standard labelled tapes does provide a rigorous algorithm for recording multi-volume files, while the interpretation of end-of-volume on a multi-volume unlabelled tape may be ambiguous. We will give an algorithm for handling end-of-volume on an unlabelled tape, but its universal acceptability cannot be assured at this time. If the EOT marker is seen while writing a data record, then write some blank tape, write at least two tapemarks, and open the new volume. But if the EOT marker is seen while writing a tapemark, then backspace over the mark and over the preceding data record, reread the data record, backspace again, write blank tape until definitely past the EOT, and write at least two tapemarks. Then get the new volume, write the last data record on it, and finally, write out the tapemark. The purpose of this complex procedure is to prevent the writing of a tapemark as the first record of a volume.

3. The header record.- Header records describe the structure and coordinate systems of the data array and convey any auxiliary parameters and accompanying text. The definition of such records is the central problem in the creation of an interchange format. With FITS, we believe that we have defined a header structure that is flexible and self-defining so that no one person will need to be responsible for authorizing and coordinating future additions to the header parameter specifications. Room is left for almost indefinite expansion. Card images in ASCII code are used in FITS to allow easy interpretation of the header records by humans and computers. The use of comments is encouraged both on the standard parameter card images and on comment cards that can transmit unlimited amounts of text, including, for example, source programs. No limit is placed on the number of card images that may be supplied. The last header record will be determined solely by the occurrence of the keyword END.

The basis grammar of a FITS header card image is given by :

keyword = value / comment

The keyword is an 8-character ASCII string left justified in columns 1-8, the = sign occurs in column 9, and column 10 is blank. With some restrictions, the value field will be written following the rules of ANSI FORTRAN 77 for list-directed input (Brainerd, 1978). However, in order to simplify the decoding of parameters on modest computers, the use of a fixed format for the most essential parameters is required and the use of the fixed format for all other parameters is recommended. The slash is optional but recommended, and is used to indicate that the remaining characters on the card constitute a comment. The slash may occur in any column after the parameter value, but at least one blank separator column is highly recommended. Lower case ASCII characters may be used in the comment field but may not appear in the keyword and value fields. The recommended, and partly required, fixed formats are as follows :

1) Logical variable : T or F in column 30.

2) Integer variable : right justified in columns 11-30 ; imaginary part, if any, right justified in columns 31-50.

3) Real variable : decimal point required ; right justified (if exponential notation used) in columns 11-30 ; imaginary part, if any, right justified (if exponential notation used) in columns 31-50.

4) Character variable: normally 8 characters in length although longer strings are allowed ; represented by a ' in

column 11, followed by the string, followed by a close 'that cannot occur before column 20.

A modest initial set of keywords is discussed in sections 4 and 5.

4. The minimum FITS data file.- In order to facilitate the decoding of normal FITS tapes by even the most basic computer systems, we have defined a minimum set of keywords which are absolutely necessary to describe the image on tape. These keywords occupy a fixed order in the header records and must use the fixed formats described in the previous section. The five+ keywords so defined are SIMPLE, BITPIX, NAXIS, NAXIS*n*, and END. The first four+ of these must occur in order as the first four+ card images in the first header record of each tape file. They carry the meanings :

SIMPLE	Logical variable, specifies whether the file conforms to the basic FITS standard
BITPIX	integer variable, specifies the number of bits used to represent each pixel value
NAXIS	integer variable, specifies the number of coordinate axes in the image
NAXIS1	integer variable, specifies the number of pixels along the most rapidly varying axis within the array
NAXIS2	integer variable, specifies the number of pixels along the second most rapidly varying axis within the array

etc.

The fundamental keyword parameter SIMPLE is required to be in the first card image of the first header record of each file. This parameter is a logical variable and indicates whether or not the present tape file conforms to the basic FITS standard. SIMPLE=T states that the array data are written as 8-bit unsigned or 16 or 32-bit twos-complement binary integers in the canonical byte order, that the array is a conventional, *n*-dimensional, regularly spaced array, and that the required keyword parameters have been provided in the specified locations in the first header record. Note that the presence of additional records that follow the image data records and/or the use of zero as the size of the image are not violations of SIMPLE=T.

The NAXIS*n* parameter is specified only up through NAXIS*m* where *m* is the value assigned to NAXIS. The limit is NAXIS999. The END keyword occupies the last card image to be used within the last header record in the file. Any remaining card images within that last header record must be filled with ASCII blanks.

All recipient programs should be coded to detect and handle cases in which there is no data array. The fundamental keywords described here will signal such cases either by setting NAXIS=0 or by setting one or more of the NAXIS*n*=0. The latter approach may well be used to describe the structure of special data records such as spectra observed at irregularly spaced positions on the sky. In such cases simple programs probably should just list the header and print a count of the number of special data records seen before the terminating tapemark.

As an example, let us suppose that we wish to export a 190x244 CCD camera image that has only 12 meaningful bits per pixel. The naked matrix contains no information about orientation on the sky or even the handedness of the coordinate system. All we know is that the first subscript, which varies most rapidly in the linear listing of the numbers of the matrix, has 190 possible values. In other words, the FORTRAN declaration for the matrix could be INTEGER *Z* (190, 244). The 16-bit, twos-complement mode can be used to transmit these data. The minimum FITS header, which is all that is known about the image, will contain only six non-blank card images which will appear in the order and with the fields positioned as shown below.

```
COLUMN            1111111111222222222233333333334
NUMBERS  1234567890123456789012345678901234567890

Card  1  SIMPLE  =                    T / comment
Card  2  BITPIX  =                   16 /comment
Card  3  NAXIS   =                    2 /comment
Card  4  NAXIS1  =                  190 /comment
Card  5  NAXIS2  =                  244 /comment
Card  6  END
```

The remaining 30 card images of this first and only header record would be filled with ASCII blanks. After the header record is written, the data matrix is written with 1440 16-bit pixel values per record. The first pixel value, *Z* (1,1), is placed in the first two bytes of the record, with the most significant bits of the value in the first byte. Then the most significant bits of the second pixel value, *Z*(2,1), are placed into the third byte and its least significant bits into the fourth byte. The process is continued in a similar manner. The most significant bits of the first pixel value in the second row, *Z* (1,2), occupy the 381st byte of the record and pixel value number 110 of row number 8 will go into bytes 2879 and 2880 of the record. Then the record is written and the most significant bits of *Z*(111,8) are placed in the first byte of the next record. The thirty-third and last data array record will contain only 280 pixel values and is filled out with zeros. Including the header, the tape file will contain 34 records and will be terminated with a tape mark. More then one file may be written on the tape and, after the last file, at least one extra tape mark is written.

SIMPLE=F implies that the tape does not conform to the basic rules in some significant way, usually because the number of bits per pixel and/or the representation of the pixel values are nonstandard. The usage of SIMPLE=F may be used for data storage or interchange within an institution or between users with similar interests and computers in those cases in which convenience overwhelms the need for standardization, but in which the outward form of this standard is still convenient to implement. We do not want to encourage any such departures from the basic FITS standard, but, FITS has been deliberately designed to have this flexibility, and the developers rely on the good sense of the community to use this power wisely.

5. Optional header parameters.- Because of the historic difficulties encountered in the interchange of image data, most images have been exchanged with no more usable information than that shown in the minimal example. However, this is really not adequate. Coordinate information and auxiliary parameters are important for unambiguous interpretation of the digital image, particularly when the object of the exchange is the intercomparison of sources as seen by various imaging detector systems. Consider, after all, that without some coordinate information the recipient cannot even know which side of the image should be displayed as "up". It is the purpose of FITS to facilitate and encourage the specification of detailed intensity, coordinate, and documentary information with each image. To this end, we have chosen a modest set of keyword parameters, the use of which in FITS headers must be regarded as optional but highly recommended. These keywords are listed in detail in Appendix A.

The remainder of this section will be devoted to a consideration of two examples of fairly complete FITS headers. The computer listings of these headers are shown in figures 1 and 2. The header shown in figure 1 contains so much information that it required two records. The first seven cards of the header constitute the required information on the map image. They appear in the specified order and give the minimum parameters needed just to read the image data records. Cards 8, 13, 34 and 36 have a blank keyword which, along with the keywords COMMENT and HISTORY, states that card columns 9-80 constitute a "comment" and may be ignored. In this case the author has inserted the blanks simply to make the header easier to read. Cards 9-12 define the image brightness scale and name. Cards 14-33 provide data on the coordinate axes, and card 35 and the following cards provide documentation. The END card is the sixth card of the second header record in the first example. The remaining 30 card images of that record are blank, and the image data begin in the third record of the tape file. The order in which the optional keywords are specified is arbitrary, and no preference for some particular pattern is to be inferred from the use of these examples. The author of these examples has chosen to provide extensive explanations of his parameter cards by adding a comment field to each one. Although this is obviously a useful practice, it is certainly not required.

In figure 1, the author has used the fixed formats defined in section 4 on cards 1-7, as required, and also on the other parameter cards, as recommended but not required. He has right justified to column 30 all logical and integer values. For floating-point numbers, both F and E FORTRAN formats are allowed and used in the example. To meet the standards of the fixed format, the decimal points must appear, and the E format items must be right justified to column 30. Character string variables are delimited by single quotes (ASCII code : 47 octal). The author of the example has met the standards of the fixed format by writing the opening quote in column 11, appending trailing blanks, and placing the closing quote in (or after) column 20. Since one cannot allow arbitrarily long strings, it was decided that recipients of FITS tapes should be prepared to interpret the first eight characters of a string (appearing between single quotes). Originators of FITS tapes may generate longer strings, but should no expect recipients to decode more than the first eight characters (e.g., the ORIGIN card in the example). Note that string notation treats leading and trailing blanks differently : 'LL   ' is equal to 'LL' but does not equal '   LL'.

This document has probably overemphasized the use of fixed formats. We have done so out of concern for potential users who possess very modest computing facilities. However, the fundamental FITS rule is that parameter values must be written in notation consistent with list-directed read operations in the ANSI FORTRAN 77 standard. Since compilers for this new standard are not yet widespread, we chose to make all the basic parameters simple and to recommend the use of fixed formats. Eventually users of FITS will probably employ the full range of flexibility allowed by FORTRAN 77, including dimensioned parameters, default-indicating commas, and a wide range of free formats. For this reason, the notations adopted in FITS (e.g., the single quotes surrounding strings and the slash to terminate the field) were chosen to be consistent with those used in FORTRAN 77.

Let us now analyze the contents of figure 1 in more detail. The required cards inform us that the tape file conforms to the basic format (SIMPLE=T), that the matrix is composed of 16-bit, twos-complement binary integers (BITPIX=16), and that it has four dimensions (NAXIS=4). The NAXISn cards appear in the required order of ascending n and show that the matrix consists of two 512x512 arrays. The thirds axis, which has a dimension of only 1, is being used with the various coordinate axis specification keywords to transmit a parameter associated with the data. In this case, it is the observing frequency (CTYPE3 = 'FREQ', CRVAL3 = 4.8856E+9 or 4.8856 GHz). The fourth dimension is a peculiar axis type difined in Appendix A. The author's comment on the NAXIS4 card is designed to remind the recipient of this definition.

The BSCALE and BZERO parameters are used to convert the image as recorded on tape into real values. The formula given in the comments field of the BSCALE card is a part of the FITS standard : the true floating-point pixel value is to be computed as the integer on the tape times the value of BSCALE plus the value of BZERO. Thus the highest real pixel value which could occur (tape value 32767) in this example is 4.19 mJy/beam area. The keywords BUNIT and OBJECT are character string variables and convey the units used for image brightness and the name of the image, respectively. A possible keyword not used in the example is BLANK. This parameter provides the integer pixel value (on tape) which indicates that the value of that pixel is unknown. For 16-bit pixels, the usual value of BLANK is -32768. The radio maps described in figure 1 contain no "blank" pixels, and no BLANK card is given.

The coordinate specification is based on the concept of a reference pixel. We give the reference pixel location along the axis (CRPIXn), the physical coordinate value (CRVALn) for that pixel, and the physical coordinate spacing between pixels (CDELTn). The reference pixel location value is based on a counter which runs from 1 to NAXIS n for each axis in the usual FORTRAN sense. CRPIXn is, however, a floating-point variable, and the reference pixel location does not have to correspond to an actual pixel. Values such as 129.5 or -3 are perfectly acceptable. CDELT n is the change in the physical coordinate as the FORTRAN counter increases by 1. In figure 1, the reference pixel occurs near the center of the array (w.r.t. the first two coordinates) and has a value of $08^h10^m10^s.07$, + 66°35'58".2. The signs of CDELT1 and CDELT2 state that the first pixel value to be found on the tape occurs at the northeast corner of the maps. The CTYPEn parameters show that the coordinate system is the (L, M) system often used in synthesis mapping. The rotation parameters CROTA n describe a coordinate system which is rotated with respect to the normal (specified) coordinate system. Users of this option should provide extensive explanatory comments. The third axis, which has dimension 1, is used simply to specify the observing frequency. The fourth axis is an arbitrary coordinate called Stokes which is defined to have values 0, 1, 2, 3, and 4 which signify beam, I, Q, U, and V Stokes polarization parameters. Thus, in figure 1, the first 512x512 array is a map of the synthesized beam and the second is a map of the unpolarized radiation field of the object. The order of the coordinate definition cards is arbitrary and all five cards are not required to be present even if some are present.

The character-string valued keywords INSTRUME, TELESCOP, OBSERVER, and ORIGIN are used to document the observing receiver/recorder, telescope, and scientist and the institution which wrote the tape. Note that the two long keywords, INSTRUMEN and TELESCOP are names which have been truncated after 8 characters. We suggest this as a good convention for the creation of new keywords. Although HISTORY is defined solely to be a comment, it is intended to be used, as in figure 1, to document processing steps which have been applied to the data. Keywords beginning with the generic string DATE are used to document various relevant dates. Note that date value strings will be coded in the form 'dd/mm/yy' (which is as good a system as any of the others) and that comments are essential when using non-standard keywords (such as DATE-MAP in the example).

To indicate the power and flexibility of the FITS system, we present an example of how a series of microdensitometer scans of a long slit spectrum of a galaxy could be stored on tape. Grids of optical or radio spectral data could be stored in a similar manner and the grid could consist of as little as one point. This example is shown in figure 2. From this header it can be seen that we have a tape containing a series of spectra of the galaxy NGC4258. The first axis is along the spectra (1024 points). The second indicates that 33 points along the slit were scanned (separated by .2 mm).

The data have been stored in a logarithmic form to allow a greater dynamic range than would be permissible with just 32 bits. This same method can be used to store any values which may have a range exceeding the number of bits available. The comment fields have been used to give additional information about the position of the slit and its orientation on the sky. The use of these comments allows the sender to provide a complete description of the data being sent. The last few cards provide information about the instrument, observer and the status of the data.

6. Suggestions to users of fits.- The preceding text has contained a variety of remarks designed to encourage the users of FITS to employ the considerable power of the system cautiously and discretely. We would also like to encourage tape writing institutions to consider the capabilities of the intended recipient institutions, particularly in the design of the coordinate systems to be used on FITS tapes. To cite a worst case, the VLA will be capable of producing spectral line maps that have numerical dimensions of 4096 x 4096 x 256. There are two scientifically useful orderings for these data : *L*-*M*-*V* (for velocity) and *V*-*L*-*M*. The VLA, which may be the only institution capable of transposing such a matrix, should offer *L*-*M*-*V* ordered maps to institutions oriented toward two-dimensional, single-frequency mapping, and *V*-*L*-*M* ordered maps to institutions oriented toward the processing of spectra. Of course, the most important thing is to use the tools of the FITS header to give the recipient a precise description of the data file.

In general, most aspects of FITS have been designed with the average computer and image processing software architecture in mind. One area in which we have deviate is the decision to pack the data into the records with maximal efficiency without regard for the commensurability of the row length and the record length. The algorithm for packing row oriented data into such records is straightforward. Inside the inner DO loop, which counts along a row, one must increment the index to the output array. Whenever the index exceeds the maximum value (720, 1440, or 2880 depending on BITPIX), the array is written and the index is reinitialized. Other, somewhat more efficient algorithms may also be constructed.

Programs that process FITS tapes can be designed simplyto skip over any files which do not conform to the basic standards. Alternatively, more sophisticated programs can analyze the keywords supplied to determine if the file falls within their present capabilities. In fact, no reasonable program will be able to handle all the possible variations of parameters allowed under FITS. Each cooperating institution should be able to handle standard files which provide less than or about the same image description parameters as are provided in that institution's internal format. The degree to which these institutions provide support for more complicated images clearly remains as a decision to be made by the institutions themselves. Obviously, the fact that the header information is supplied in card image form, preferably with comment information, is a great asset and means that extensions to the FITS standard can propagate through the community as tapes are circulated. Perhaps someone, such as the IAU Commission IX working group on Data Processing in Astronomy, will assume the role of czar to supervise the scheme. Until that happens we will all have to be careful to avoid compromising this tool by careless departures from the basic FITS standard.

Additional problems are faced by the programmer of a computer that does not have a word size matched to the value of BITPIX. However, the elegant universal commensurability of the record length makes these problems fairly simple to solve. Consider the case of a computer that has a word size of 36 bits. When a record is in the memory, it will occupy exactly 640 memory cells, but there is an awkward placement of the pixels within the words of the memory. The programmer needs to construct subroutines to unpack nine 16-bit pixels from four 36-bit words, nine 32-bit pixels from eight 36-bit words, and nine 8-bit pixels from two 36-bit words. The whole record can be unpacked by calling the appropriate routine in a loop of suitable length (e.g., 160 calls for a 16-bit unpacking).

The programmer must also consider how to handle twos-complement 16-bit and 32-bit pixels. For example, when a twos-complement 16-bit integer is unpacked on a machine of word length greater than 16 bits, the result will be a right-justified integer between 0 and 65535 (177777_8). If the number is greater than 32767, it can be given its proper negative value by subtracting 65536 (200000_8). This operation will work correctly on both twos-complement and ones-complement computers. A similar scheme applies to 32-bit pixels (subtract 40000000000_8). If the computer's word size is less than the value of BITPIX, some compromise will be required, and the chosen solution is likely to depend on the application.

Users who have difficulty with blocks missed during reading can insert an extra column in the array as a checksum or row number and explain its presence in the header. In general this should not be necessary.

7. Conclusions.- A format for the interchange of astronomical images on magnetic tape has been developed and is already in use at several major observatories. This FITS format provides a single, flexible, and powerful mechanism for the transmission of *n*-dimensional, regularly spaced data arrays together with an unlimited number of parameters which may be used to describe and document the arrays. These parameters are written in a form which is easily interpreted by both humans and computers. A basic set of keywords for such parameters has been devised, but the addition of new keywords as they are needed poses no difficulty. The system is so flexible that no institution can expect to be able to decode all possible FITS tapes. Receiving institutions should encounter no difficulty in simply listing the FITS headers and should be able to decode normal data forms related to their application. Transmitting institutions should write the tapes, whenever possible, in a way which will aid the receiving institution.

The FITS coordinate definition scheme is obviously quite flexible and general, but it does contain some pitfalls. The most serious of these is beyond the scope of this paper and involves the need to provide precise definitions of the standard coordinate types. For example, we must ultimately determine standards for the senses of the Stokes Q, U, and V parameters and for the spatial coordinates to which Q and U refer. Other problem coordinates include the form of the projective geometry for (L, M) and the formula for computing velocity (e.g., $\Delta\lambda / \lambda$ vs. $\Delta\lambda / \lambda_0$). Readers are encouraged to send proposals for resolving such problems to one of the authors.

Appendix A

the FITS format in outline

I. Purpose

A. Communication of *n*-dimensional data arrays between institutions.

B. Communication of detailed parameters describing such arrays.

C. Communication of unlimited quantities of text.

D. Avoid impact on formats currently in use.

II. Physical form

A. Medium

1. ANSI standard magnetic tape
2. Either labeled with ANSI standard labels or unlabeled (preferred)
3. Seven-track allowed, nine-track preferred

B. Data records

1. All records shall be 23040 bits in length
2. Order within each 'image'

a. One or more "header" records
b. Zero or more data array records
c. Zero or more "other" records
d. One tapemark

3. Basic forms

a. Header records : 2880 8-bit characters
i. In 7-bit ASCII code
ii. Organized as 36 80-character card images
b. Data records : binary integers are standard
i. 8-bit unsigned
ii. 16-bit twos complement
iii. 32-bit twos complement
c. Other records : not currently defined
d. Tapemark : ANSI standard

C. Data

1. Bits are ordered within bytes following the ANSI standard

2. Bytes are sequenced within tape records

a. For characters in strings (i.e., header records), in the order in which they are to be parsed.
b. For bytes within pixel values or other binary data in order of decreasing significance (i.e., sign bit first).

3. Pixel values occur in the standard FORTRAN sequence with

a. 2880 pixel values/record if BITPIX = 8
b. 1440 pixel values/record if BITPIX = 16
c. 720 pixel values/record if BITPIX = 32

4. Example : four-dimensional array IMAGE (M1, M2, M3, M4)

a. Declare NAXIS = 4, NAXIS1 = M1, NAXIS2 = M2, NAXIS3 = M3, NAXIS4 = M4 (see IIIC).
b. Place pixel values on tape in order : IMAGE (1,1,1,1) through IMAGE (M1,1,1,1) followed by IMAGE (1,2,1,1) through IMAGE (M1,2,1,1) etc.
c. Pack pixels within data records with maximal efficiency, paying no attention to commensurability of row length (M1) with record length (2880, 1440, or 720)

5. The first data array record, if any, follows the last header record.

6. Unused pixel value positions following the array values within the last data array record are filled with zeros.

III. <u>Format of header record card images</u>

A. Basic sequence

1. Keyword - left justified in columns 1-8

2. ' = ' - in columns 9-10 (except comment cards)

3. Parameter value - in ANSI FORTRAN 77 list - directed read format anywhere in columns 11-80

4. '/' - optional, terminates value field

5. Comment - follows the /

B. Format

1. Character code : ASCII

a. Codes allowed before / : "Field Codes", "Non Alpha"
b. Codes allowed after / : as above plus "Lower Case"

2. Fixed format types

a. Logical - character T or F in column 30
b. Integer - right justified in columns 11-30, complex part (if any) right justified in columns 31-50
c. Floating point - columns 11-30, complex part (if any) in columns 31-50
i. F notation is free format within the specified columns, but the decimal must appear.
ii. E notation is right justified in the specified columns.
d. Character - surrounded by character '(octal 47)
i. Opening' in column 11
ii. Closing' in or after column 20
iii. Heading blanks are significant, trailing blanks are not.

C. Required keywords - must appear in each FITS file

1. The first 3+ card images in the first header record of each file must use the following keywords with fixed format in the order given here.

a. SIMPLE (logical) file conforms to basic format ?
b. BITPIX (integer) # bits used on tape for each pixel value
c. NAXIS (integer) # axes in array
d. NAXIS1 (integer)# pixels on fastest varying axis
e. NAXIS (integer) # pixels on second fastest varying axis
f. NAXIS3 (integer)# pixels on third fastest varying axis
g. NAXIS999 (integer) # pixels on 999th fastest varying axis

2. If the value assigned to NAXIS is m, then only NAXIS1 through NAXISm should appear. If $m = 0$, then no NAXISn cards should appear.

3. The last (logical) header card image within the last header record of each file must use the keyword END. Following card images within that header record must be blank.

D. Optional keywords - use and ordering of the following keywords is optional.

1. BSCALE (floating)	Scale factor used to convert tape pixel values to true values (true = [tape x BSCALE] + BZERO).
2. BZERO (floating)	Offset applied to true pixel values (see above formula).
3. BUNIT (character)	Brightness units (see list of recommended values below).
4. BLANK (integer)	Tape pixel value assigned to undefined-value pixels.
5. OBJECT (character)	Image name.
6. DATE (character)	Date file written ('dd/ mm/ yy').
7. DATE-OBS (character)	Date of data acquisition ('dd/mm/yy').
8. ORIGIN (character)	Tape writing institution.
9. INSTRUME (character)	Data acquisition instrument.
10. TELESCOP (character)	Data acquisition Telescope.
11. OBSERVER (character)	Observer name/identification.
12. BLANK (none)	Cols. 9-80 are a comment.
13. COMMENT (none)	Cols. 9-80 are a comment.
14. HISTORY (none)	Cols. 9-80 are a comment.
15. CRVAL n (floating)	Value of physical coordinate on axis n at the reference pixel (see notes on units below).

16. CRPIX *n* (floating)	Array location of reference pixel along axis *n*.
17. CDELT*n* (floating)	Increment in physical coordinate along axis *n* as FORTRAN counter increases by 1.
18. CTYPE *n* (character)	Type of physical coordinate on axis *n* (see list of recommended values below).
19. CROTA*n* (floating)	Rotation angle of actual axis *n* from stated coordinate type.
20. DATAMAX (floating)	Maximum data value in the file.
21. DATAMIN (floating)	Mimimum data value in the file.
22. EPOCH (floating)	Epoch of coordinate system (years).

E. Suggested values for BUNIT

1. 'K '	Kelvins
2. 'JY/BEAM'	Jy per beam area
3. 'JY/PIX '	Jy per pixel
4. 'MAG/PIX '	Magnitudes per pixel
5. 'M/SEC '	Meters per second (e.g., velocity fields)
6. 'DEGREES'	Angles in degrees (e.g., polarization position angle)
7. ' '	Unitless (e.g., optical depth, spectral index)

F. Suggested values for CTYPE *n*

1. 'RA '	Right ascension in degrees
2. 'DEC '	Declination in degrees
3. 'LL '	Tangent plane (E-W) in degrees
4. 'MM '	Tangent plane (N-S) in degrees
5. 'GLON'	Galactic longitude in degrees
6. 'GLAT '	Galactic longitude in degrees
7. 'ELON '	Ecliptic longitude in degrees
8. 'ELAT '	Ecliptic latitude in degres
9. 'TIME '	Time in seconds
10. 'FREQ '	Frequency in Hertz
11. 'LAMBDA '	Wavelength in meters
12. 'VELO '	Velocity in meters/second
'VELO-LSR'	Velocity wrt local standard of rest
'VELO-HEL'	Velocity wrt Sun
'VELO-OBS'	Velocity wrt observer
13. 'PIXEL '	Unitless
14. ' '	Unitless
15. 'STOKES '	Stokes polarization parameters and synthesized beam pattern, has values 0 beam, 1=*I*, 2=*Q*, 3=*U*, 4=*V*
16. 'COMPLEX'	Complex valued, has values 1 = real, 2 = imaginary
17. 'DISTANCE'	A distance on the sky between points. This is useful for non-standard coordinate systems
18. 'ANGLE '	An angle on the sky in degrees

IV. Units

A. Basic set

1. Consistent with the International System of Units (SI)
2. Add degrees for angles
3. Add Janskys for flux

B. Examples

1. Distances in meters
2. Masses in kilograms
3. Times in seconds
4. Temperatures in Kelvins

References

BRAINERD, W. : 1978, *Comm. ACM*, 21, 806 ; 1977, 'Workshop on Standards for Image Pattern recognition', NBS Publication 500-8, U.S. Department of Commerce.

```
      000000000111111111122222222223333333333444444444455555555556666666666777777777778
CARD# 12345678901234567890123456789012345678901234567890123456789012345678901234567890

1/ 1: SIMPLE  =                    T          / BASIC FORMAT FROM NRAO(CV)
1/ 2: BITPIX  =                   16          / 2-BYTE TWOS-COMPL INTEGERS
1/ 3: NAXIS   =                    4          / NUMBER OF AXES
1/ 4: NAXIS1  =                  512          / # PIXELS/ROW  (RA)
1/ 5: NAXIS2  =                  512          / # ROWS        (DEC)
1/ 6: NAXIS3  =                    1          / # FREQUENCIES
1/ 7: NAXIS4  =                    2          / # STOKES (BEAM, I,Q,U,V)
1/ 8:
1/ 9: BSCALE  =        1.278419E-07          / REAL = TAPE*BSCALE + BZERO
1/10: BZERO   =                  0.0          / NO BIAS ADDED
1/11: BUNIT   = 'JY/BEAM '                    / UNITS OF BRIGHTNESS
1/12: OBJECT  = '0810+665'                    / SOURCE NAME
1/13:
1/14: CRVAL1  =          122.5419617          / REF POINT VALUE DEGREES
1/15: CRPIX1  =               256.00          / REF POINT PIXEL LOCATION
1/16: CTYPE1  = 'LL      '                    / COORD TYPE: VALUE IS RA
1/17: CDELT1  =        -6.944167E-05          / COORD VALUE INCREMENT WITH COUNT DEGR
1/18: CROTA1  =                  0.0          / CCW ROTATES -RA INTO +DEC
1/19: CRVAL2  =           66.5995040          / REF POINT VALUE DEGREES
1/20: CRPIX2  =               256.00          / REF POINT PIXEL LOCATION
1/21: CTYPE2  = 'MM      '                    / COORD TYPE: VALUE IS DEC
1/22: CDELT2  =        -6.944167E-05          / COORD VALUE INCREMENT WITH COUNT DEGR
1/23: CROTA2  =                  0.0          / CCW ROTATES +DEC INTO +RA
1/24: CRVAL3  =       4.8856000E+09           / REF POINT VALUE HZ
1/25: CRPIX3  =                  1.0          / REF POINT PIXEL LOCATION
1/26: CTYPE3  = 'FREQ    '                    / COORD TYPE: VALUE IS HZ
1/27: CDELT3  =                  0.0          / COORD VALUE INCREMENT WITH COUNT DEGR
1/28: CROTA3  =                  0.0          / FREQ ROTATION UNDEFINED
1/29: CRVAL4  =                  0.0          / REF POINT VALUE STOKES #
1/30: CRPIX4  =                  1.0          / REF POINT PIXEL LOCATION
1/31: CTYPE4  = 'STOKES  '                    / COORD TYPE: VALUE IS #
1/32: CDELT4  =                  1.0          / COORD VALUE INCREMENT WITH COUNT ST #
1/33: CROTA4  =                  0.0          / STOKES ROTATION UNDEFINED
1/34:
1/35: INSTRUME= 'VLA     '                    / NRAO(CV) VLA MAPPING.PROGRAMS
1/36:
2/ 1: DATE-MAP= '16/10/78'                    / MAP CREATION DATE DD/MM/YY
2/ 2: DATE    = '27/05/79'                    / MAP WRITING DATE DD/MM/YY
2/ 3: ORIGIN  = 'NRAO(CV) PGM=DEC2FITS(V1)'
2/ 4: HISTORY VLACV MAP   METHOD='FFT' DATA='OBS. VISIBILITY'
2/ 5: HISTORY VLACV MAP   WCONU=  0.00000E+00 WCONV= 0.000000E+00 TCONU= 0
2/ 6: END
```

FIGURE 1.- A FITS header

```
      000000000111111111122222222223333333333444444444455555555556666666666777777777778
CARD# 12345678901234567890123456789012345678901234567890123456789012345678901234567890

1/ 1: SIMPLE  =                    T          /
1/ 2: BITPIX  =                   32          / 4 BYTE TWOS-COMPL INTEGERS
1/ 3: NAXIS   =                    4          / NR. OF AXES
1/ 4: NAXIS1  =                 1024          / NR. PTS. PER SPECTRUM
1/ 5: NAXIS2  =                   33          / NR. SAMPLES ALONG THE SLIT
1/ 6: NAXIS3  =                    1          / R.A. OF CENTER SAMPLE
1/ 7: NAXIS4  =                    1          / DEC. OF CENTER SAMPLE
1/ 8:
1/ 9: BSCALE  =               1.0E-6          / REAL=TAPE*BSCALE + BZERO
1/10: BZERO   =                    0          / ARBITRARY ORDINATE
1/11: BUNIT   = 'LOG F   '                    / FLUX SCALE
1/12: OBJECT  = 'NGC 4258'                    / NAME OF OBJECT
1/13:
1/14: CRVAL1  =        4861.342E-10           / H BETA WAVELENGTH
1/15: CRPIX1  =                 37.5          / LOCATION OF H BETA LINE
1/16: CRTYPE1 = 'LAMBDA  '                    / NAME OF AXIS
1/17: CRDELT1 =              3.2E-10          / RESOLUTION BETWEEN POINTS
1/18:
1/19: CRVAL2  =                  0.0          /
1/20: CRPIX2  =                   17          / CENTER IS NR. 9
1/21: CRTYPE2 = 'DISTANCE'                    / ALONG THE SLIT ON THE PLATE
1/22: CRDELT2 =               0.0002          / 2 MM BETWEEN SAMPLES ALONG SLIT
1/23:
1/24: COMMENT                                   THE R.A. OF THE SLIT CENTER IS 184.1225 DEG.
1/25: COMMENT                                   THE DEC. OF THE SLIT CENTER IS 47.58139 DEG.
1/26:
1/27: COMMENT                                   THE SLIT IS ORIENTED AT AN ANGLE OF
1/28: COMMENT                                   35.4 DEGREES. ROTATION IS CLOCKWISE N THRU W
1/29:
1/30: COMMENT                                   THE PLATE SCALE IS 18.7056 DEGREES PER M
1/31: COMMENT                                   THIS IS THE SAME AS THE PALOMAR SURVEY
1/32:
1/33: COMMENT THE DISPERSION WAS 40x10E-10 M PER 10E-3 METER
1/34:
1/35: INSTRUME= 'ABC CAMERA'
1/36: TELESCOP= '90 INCH '
2/ 1: OBSERVER= 'SMITH   '
2/ 2: ORIGIN  = 'KPNO-PDS'
2/ 3: DATE    = '27/10/78'
2/ 4: HISTORY DATA CORRECTED FOR DARK FIELD BUT NO ABSOLUTE INTENSITY CAL.
2/ 5: COMMENT SEEING 2 ARC SECONDS
2/ 6: END
```

FIGURE 2.- A FITS header for spectral data.