Reprinted from

Eleventh International Symposium

Machine Processing of

Remotely Sensed Data

with special emphasis on

Quantifying Global Process:
Models, Sensor Systems, and Analytical Methods

June 25 - 27, 1985

Proceedings

Purdue University
The Laboratory for Applications of Remote Sensing
West Lafayette, Indiana 47907 USA

A NEW FORMAT PROPOSAL

FRED C. BILLINGSLEY

Jet Propulsion Laboratory
California Institute of Technology
Pasadena, California

BACKGROUND

The American Congress on Surveying and Mapping (ACSM) National Committee on Digital Cartographic Data Standards has been chartered by the National Bureau of Standards to recommend a format for the interchange of digital cartographic data. The committee presented the following format concept (1) at the spring 1985 ACSM meeting in Washington, D.C. It is the intent that this concept be reviewed by the concerned public, who are asked to provide reactions to the committee.

PHILOSOPHY AND CONSTRAINTS

The interchange format must be able to carry all types of cartographic data - point and polygon data, raster (image) data, features and attributes, and the necessary topological relations. While it is desirable to define a single format, it was recognized that the diversity of data types prevents this. A format family is therefore necessary.

The necessity for a family brings the necessity for conveying to the receiving computer the logical structure of the data being presented. This has been done in the past by paper documents and their corresponding individual logging software. Modern practice recognizes the difficulties encountered with this approach when the format changes or different formats are being received, and has begun to embody some form of data definition language (DDL) as part of the data set. This proposed format family also includes several styles of DDL.

The committee recognizes the large number of data files currently in the numerous archives, and the necessity for their interchange. A two-step process toward the development of a recommended family has been adopted: The first step allows the transmission of presently-existing files without modification, with the inclusion of the necessary format information as part of accompanying core data. The second step is the setting forth of preferred formats for each of the data types involved, again with the format information included as a DDL. In the present proposal, the first step only is defined; ongoing work will define the preferred formats during 1985 and 1986.

DATA DEFINITION STRUCTURE

There are several distinct ways in which data definition information can be recorded, all following a general Tag-Length-Position sequence. Two methods are allowed in the present definition: an in-line definition of the sturcture of the data fields in the same record; and a record set consisting of a Data Definition Record and its defined Data Records.

The structure is based on the International Standards Organization (ISO) DIS 8211 data distribution format. This format has been under development since 1978 in the Department of Energy, and is now a proposal to ANSI, and has been recently adopted by the ISO. The 8211 scheme is record based, with Data Definition and Data records. The components and their uses are given in Table 1.

As the 8211 format is record based rather than file based, certain features necessary for transmission of an entire record group are missing or clumsy. In particular, needed are the capability to indicate the grouping and uses of the various types of records, the ability to define and include records of various formats within one file, the ability to enclose and identify already existing data sets in formats under other authority control (such as the USGS DLG), and to indicate that authority.

This proposed ASCM format attempts to provide these capabilities in a manner

Record	Component	Function
DDR	Leader	Identifies the DDR
	Directory	Contains the entry map (sizes of the tag, length, and position fields of the corresponding directory entries in this record) Gives Tag, length, and position (relative to start of the Data Descriptive section) of each Data Descriptive field in this record
	Data Descriptions	Structure of each corresponding Data Field in the DR. (This is the "data" of this record.)
DR	Leader	Identifies the DR
	Directory	Contains the entry map (sizes of the tag, length, and position fields of the corresponding directory entries in this record) Gives Tag, length, and position (relative to start of the Data section) of each data field in this record.
	Data Fields	Have the structure described in the corresponding DDR Data Descriptive fields.

TABLE 1 - Components of the DIS 8211 Format

which retains the essence of the DIS 8211 proposal. Structures and procedures of the 8211 are used wherever possible. Certain of the 8211 segments are used in data records for data identification when the entire 8211 structure is not desired.

In electronic transmissions in an open environment, confusion may result at the receiving terminal between binary information and control codes. To avoid this potential problem and to provide unambiguous transmissions, the format shall contain only ASCII characters in all locations except data fields. Binary data shall be allowed in the data fields.

FILE IDENTIFICATION AND DATA DEFINITION

A record is a collection of related data and is treated as a logical unit.

A file shall consist of one or more records, at least the first of which shall contain a core (defined below), and optionally a core extension and a data section.

The format definition defines the structure of each record in the file. The basic form of the records shall be:

CORE RS [ERROR CORRECTING CODE] RS [CORE EXTENSION] RS [DATA FIELDS] RS

where RS indicates the ASCII Record Separator and [ ] indicates optional fields. Note that the RSs are definitive and must be included.

Coding in the Core and in the Core Extension shall be ASCII characters or certain ASCII control codes as called out herein. Coding in the Data section may be ASCII or Binary, as defined for the given instance in the Data Descriptive sections.

THE CORE

The CORE shall consist of a series of segments, each consisting of a two-character decimal size field which indicates the number of characters following in the field, including the separator, a data field so dimensioned, and an ASCII Unit Separator (US). The data field may be subdivided to include a Tag subfield, an ASCII Group Separator (GS), and other data. Absence of the GS shall indicate no tag. In the last segment, the Unit Separator shall be replaced by an ASCII Record Separator (RS). Following this will be an error-correcting code, followed by another Record Separator.

Recognition that this is the logical format of the file which has been read (if this is not known a priori) would be through hypothesizing that it is, and then finding the USs and the RSs in the proper places.

The Core Extension shall consist of further data definition information.

The Data fields shall have formats as described in the Core and Extension. The last data field shall terminate with an ASCII Record Separator.

CORE SEGMENTS CONTENTS

The core contains file identification and certain format information. The core definitions shall consist of the specific segment definitions and (if tags are not used) the segment order. For segments using tags, the meanings of the segments shall be determined from the contents.

The first two segments shall be the Maintenance Control Authority and its Format ID Control. These two segments shall not contain tags. The Core may be terminated at any desired point after the first two segments.

MAINTENANCE CONTROL AUTHORITY

Defines the authority under which this format is defined.

MAINTENANCE CONTROL FORMAT ID

Defines the specific format identification, including revisions as defined by the Maintenance Control Authority.

TAG LIST Tag: TAGLIST

Following the first two segments, an optional segment with the tag TAGLIST may be used to define the order of the remaining Core segments. If a TAGLIST segment is used, its data field shall consist of its tag, an ASCII Group Separator, and a list of the tags for the segments to be used, in order, separated by commas. The segments so identified may optionally have tags, or not.

If Core segment tags are not used, the remaining segments of the Core shall be, in the following order:

DATA START POSITION DSP

A decimal number. The byte position of the start of data in the record, referred to the first byte of the record.

RECORD TYPE AND SUBTYPE RTS

A four byte code using ASCII characters.

RECORD NUMBER RNUM

A decimal number. The first (physical) record in each file shall be the number "1".

RECORD LENGTH RLEN

A decimal number. The total physical record length, in bytes.

MAXIMUM RECORD LENGTH IN FILE MAXLEN **

A decimal number indicating the number of bytes in the longest record in the file.

EXTERNAL AUTHORITY EXAUTH

Under some conditions, the structure and meaning of the data fields can be determined by reference to known external documents. This field defines the source of those documents. This field shall apply to the entire file when found in the Volume Descriptor, or only to those data records carrying the same DDI, when found in a DDR.

EXTERNAL FORMAT DEFINITION EXTFMT

The document identification defined by the External Authority. This field shall apply to the entire file when found in the Volume Descriptor, or only to those Data records carrying the same DDI, when found in a DDR.

PACKING FACTOR PACK

Defines the division of logical records into physical records.

RECORD REPETITION INDICATOR RECREP

Allows subsequent data records to omit the Core. This indicator shall contain a decimal number indicating the number of subsequent data records without a Core or a Core Extension. The Data fields of those records shall be the same structure as those of the current record.

ERROR CORRECTING CODE (No Tag)

A segment of the Core containing information to allow correction of certain potential errors which may occur in the core. It shall consist of [format TBD. It is anticipated that a cyclic redundancy code using the ISO-recommended polynomial will be specified].

** Not in present definition: to be added.

THE CORE EXTENSION

The Core Extension carries format identification pertaining to the structure of the data fields. Fields in the CORE EXTENSION shall consist of one or more of the following segments:

DATA DEFINITION INDICATOR DDI

Defines the style and location of Data Definition (DD) information pertaining to the data fields in the remainder of the record.

The DDI may take one of three forms, depending on the data definition style to be used:

	DDI	Entry Map
No Data Definition	Zero	None, or all zero
Inline, Long Form	Term Symb	<> 0
Inline, Short Form	Term Symb	None, or all zero
DDR/DR Groups	Group Code	None or 0 in Core, <> 0 in 8211 Leader

The DDI Terminator Symbol shall be a non-conflicting ASCII printable symbol, but not ellipsis, comma, or asterisk, which have reserved meanings.

The DDI Grouping Code shall be an ASCII alphameric character, but not zero.

ENTRY MAP EMAP

A field (derived from the DIS 8211 Leader) describing the sizes of Tag, Length, and Position fields in the Data Directory:

Size of Dir Length Field	(=m)	Length: I1
Size of Dir Pos'n Field	(=n)	I1
Reserved	(ASCII Zero)	I1
Size of Dir Tag Field	(=t)	I1

8211 DATA DEFINITION RECORD LEADER DDRL

Identifies a DDR. Contains an entry map and certain format information. See the 8211 defining document.

8211 DATA RECORD (DR) LEADER DRL

Identifies a DR. Contains an entry map and certain format information. See the 8211 defining document.

DIRECTORY DIR

Multiple entries, one for each identified data field in the Data Description segment, as follows (styled after 8211):

Tag of the ### Field Length: At
Length of the ### Field Im
Position of the ### Field In

Data Description, in the DDR Structure Definition, in an in-line structure

DATA DESCRIPTION seg of the DDR DESC

Defines the structure of the DR data fields, and has a structure for each entry (corresponding to Directory entries), in accordance with 8211 Par 6 (q.v. for details):

Field Control	Defines the field type: Integer, Complex, etc.
Separator	RS US
Field Name	Optional, user-supplied
Label	Optional labels for sub-fields
Format Controls	Optional Format Designations as needed

INLINE STRUCTURE DEF'N - LONG FORM ILL

When a shorter form than the entire DDR/DR is desired, the data field of the ILL structure definition shall be:

Tag	Same tag as used in the Directory, followed by the DDI Terminator symbol. *
Length	Length of corresponding DR data field, followed by the DDI Terminator Symbol. *
Position	Relative Position of the corresponding DR data field in the data area (first position is zero), followed by the DDI Terminator Symbol. *
Format Control	Format Designation as defined below, followed by the RS.

A vector data field is one containing a series of major subfields. The Format Control for vector fields shall use the designated Terminator Symbol of the DDI to separate the sub-fields. Each subfield, in turn, may be divided in accordance with the format control definitions.

INLINE STRUCT DEF'N - SHORT FORM ILS

Format Control For each major data field, a Format Designation, followed by the US.

The last Format Control field shall be followed by the RS instead of the US. Vector Format Control fields shall use the DDI Terminator Symbol as internal separators between major data fields.

THE DATA SECTION

The Data Record DATA section shall consist of data fields as described in the preceding segments, followed by an ASCII Record Separator.

In records using the in-line structure or no data definition, the data fields will be contiguous and not have terminators.

The Data Definition Record DATA section is the Data Description segment.

INLINE STRUCTURE FORMAT CONTROLS

The Inline Structure Format Controls are defined to closely match those of the DIS 8211 document. This will make the Format Control callouts the same whether using the Inline definitions or the full DDR/DR structure of 8211.

The format controls specify the character-by-character or bit-by-bit structure of the data fields. The format controls are required to specify the sequence and type of the subfields in a mixed data field.

The format controls shall take the form:

{Y¦jY¦k(jY,...),...}

where

j,k are positive integers signifying the number of times the following data type or group of data types, respectively, is to be repeated.

Y implies [Z¦Z(*)¦Z(n)¦Z(m)]

Z = {A signifies character data,
{I signifies implicit-point representation,
{R signifies explicit-point unscaled representation,
{S signifies explicit-point scaled representation,
{C signifies character mode bit field,
{B signifies bit field data,
{M signifies a mask corresponding to a preceding B field *
{X signifies unused character positions and is limited to the form X(n)

{ implies the enclosed expression is to be treated as an entity for purposes of repetition and nesting,

* is an arbitrary non-conflicting non-numeric user delimiter

¦ implies an alternate choice,

(n) is a field width specification,

n is a positive integer,

(m) is a bit-field mask for an M-type specification *

m is an ASCII "0" or "1", *

... implies repetition of the preceding expresion.

The use of the format control is governed by the following rules:

a) The order of the fields and their types specified with format controls shall correspond to the data field when the format controls are traversed from left to right expanding the nested terms from the left. If the data field is not exhausted, the format shall repeat from the left hand parenthesis corresponding to the next to the last right hand parenthesis, not including those parenthesis used to delimit field width and using the associated repetition factor if any.

b) Data Record data fields shall be delimited by the ASCII US.

c) (*) implies the presence of a character, *, as a terminating user delimiter for the corresponding data subfield where * is an arbitrary non-numeric non-conflicting character. The delimiter of the last subfield of a data field shall be replaced by the ASCII RS.

d) Data fields of the I-type, S-type amd R-type will specify a decimal number in one of the following formats:

```
I-type  ::= #*[(+¦-¦#)] dd*
S-type  ::= #*[(+¦-¦#)] (dd*.d*¦d.dd*)
R-type  ::= (I-type¦S-type) E (+¦-) dd*
```

where

```
     # = space
     d = (0¦1¦2¦3¦4¦5¦6¦7¦8¦9)
                i.e., a decimal digit
     + = plus sign
     - = minus sign
     . = decimal point
```

a* implies zero or more repetitions of a
(a;b) implies a choice of a or b
[ ] implies optionality.

The implicit decimal point of the I-type is at the extreme right, the value zero may not be minus, an R-type number of the form a E b has the decimal value of a x 10exp b, and I-type and S-type have their usual decimal values.

e) C-type data fields will specify bit strings as a sequence of ASCII characters "0" and "1" correspomding to the binary digits in the bit string represented.

f) Binary data shall be specified as B-type, with the fixed field width format control specifying the (all equal) number of bits in each binary number. Arrays containing fixed-length bit data shall have each subfield adjacent to the preceding subfield. The last byte of one or a series of adjacent fixed-length bit sub-fields shall be filled on the right with binary zeros. The field shall be terminated with the appropriate field terminator.

In the event that the significant bits * do not completely fill the fixed field * width, a mask shall be supplied. The * mask shall be designated as an M-type * field specification, with the field * width portion containing ASCII * characters "0" and "1", signifying * unused and occupied bit positions, * respectively, of the binary data fields * defined by the preceding B-type field * specification. *

* designates definitions not included in DIS 8211.

DEFINITIONS

ASCII CHARACTERS

These are the characters specified as the Standard 7-bit character set of ANSI Standard X3.4-1977. These are the same as the International Reference Version (IRV) of ISO Standard 646-1983, with the exception of positions 2/04 and 7/14, which have the currency symbol and overscore, respectively, in the IRV.

BYTE

A group of binary digits (bits) used as a unit. This will be the term used to indicate an OCTET, an 8-bit byte.

FILE

A collection of records treated as a unit.

INFORMATION SEPARATORS

Information Separators are defined as control characters used to separate and qualify data logically. Their specific uses shall be as specified herein, and not in a hierarchical order. The information separators and their coded representations are specified as follows:

ASCII Acronym	Name	Coded Location	Numeric Value Decimal	Hex
US	Unit Separator	1/15	31	1F
RS	Record Separator	1/14	30	1E
GS	Group Separator	1/13	29	1D
FS	File Separator	1/12	28	1C

RECORD

A collection of related data treated as a logical unit.

USING THE FORMAT - RECORD MAKEUP

CORE Considerations

The Core may be constructed with or without tags for its components. In the absence of tags or external documentation describing the segment order, the list of segments, in the expected order as defined herein, must be followed for interpretation.

The definition allows the use of Core Segment Tags, by allowing each segment to be subdivided, with a Tag in the first division. This will make each segment: Data Length, Tag, ASCII Group Separator (GS), Data Field, Unit Separator. The Data Length shall represent the Tag, GS, and Data fields together. Lack of a GS within a segment indicates no Tag.

Alternately, one segment might be designated as a tag list, in which case, the other segments would not need tags. This shall occur early in the core, immediately after the Local Format ID. Its data field shall be subdivided with the first section being a TAGLIST tag, then the ASCII Group Separator, and variable length sub-fields, separated with commas, containing the series of Core Tags, in order. This will allow the local implementor to use any desired Tag structure, including none at all. The core TAGLIST segment list is considered to be an ordered list of core segments, which segments may optionally include the tags defined in the TAGLIST.

In all cases, the first two Core segments shall be the Local Control Authority and the Local Format ID. These segments shall not contain tags.

FILE STRUCTURE

Files may be defined and transmitted in several versions, depending upon the robustness desired and the degree of callout of the various field structures. These forms will use the components described above in various combinations:

1. No Data Definition at all

This would be coded as zero in the DDI. External documentation must be sought for all data record structures.

Major components: DCDS CORE, w/o Entry Map
DATA (UNTAGGED FIELDS)
Recognition Coding DDI: Zero
Core Entry Map Length = 0
(No Entry Map)

2. Data definition (DD) included within the Data Record

The DD information would follow the core and precede the data fields. This structure might be used when each data record has a unique structure, and/or where the overhead of the full 8211 structure is undesirable. Thus, the shortened definition is appropriate. The DCDS core and a data definition structure derived from the 8211 structure is used.

Major Components: DCDS CORE, including Entry Map
DIRECTORY
INLINE STRUCTURE
DATA (UNTAGGED FIELDS)
Recognition Coding DDI: Non-conflicting Terminator Symbol
Core Entry Map > Zero

The short form of inline data structure record is:

Major components: DCDS CORE, w/o Entry Map
SHORT FORM STRUCTURE
DATA (UNTAGGED FIELDS)
Recognition Coding DDI: Non-conflicting Terminator Symbol
Core Entry Map Length = 0
(No Entry Map)

3. Data Definition Record - Data Record Sets

Data Definition Records (DDR)

The 8211 definition calls for only one DDR per data record group, with all fields tagged in the Directory and Data Descriptions. In the DCDS structure multiple groups are allowed, with all of the records in a record group having the same format. Thus, one such DDR would precede a set of data records, DR, each of which has the structure as defined in the preceding DDR. This requires recognition coding in the DR that there is an external DDR. This will be indicated by a Grouping Code in the DDI of each DR and DDR, using the same symbol throughout a DDR-DR group.

The 8211 structure will start immediately after the core. Although there is some overlap between the purposes and information in the core and the 8211 DDR leader, the 8211 leader will be included in its entirety.

When the 8211 DDR or DR core is used, the Core Entry Map is not needed; the 8211 Entry Map (part of the Leader) shall convey the required information.

DDR major components are: DCDS CORE
8211 DDR LEADER
including ENTRY MAP
DDR DIRECTORY
DDR DATA DESCRIPTION
Recognition Coding DDI: Desired Record Grouping Code
Core Entry Map Length = 0
8211 Entry Map > Zero

For 8211 details, see the 8211 defining document.

Data Records (DR)

The entire DR leader will be included (or not), as indicated in Par. 5.3.1.3 of 8211. The DDR/DR repetitive characteristics would be coded in RP 6 of the Leader, using the coding of Par 5.3.1.3 of the 8211 document. However, note that the DR Base-Address-of-Data will be different in the DRs, depending on the repetitive (or not) inclusion of subsequent leader and directories. Therefore, the DR leader address-of-data field shall be defined to apply to the concurrent record, and the data start must be calaculated by the software for subsequent records. Data locations as given in the Directory are relative to the start of the Data area

(first position = zero).

DR major components are: DCDS CORE
8211 DR LEADER
including ENTRY MAP
DR DIRECTORY
DATA FIELDS, WITH
TERMINATORS

Recognition Coding DDI: Record Grouping
Code to match DDR DDI
Core Entry Map Length = 0
8211 Entry Map > Zero

For details see the 8211 defining document.

Data record structure is guided by several considerations: 1) It is desirable to enclose extant data sets without further modification. These normally will not have the 8211-style data field terminators. However, new data sets following the 8211 must have these terminators. 2) In the DDR/DR structure, the 8211 Record ID field must be present, with a tag 0...1. Other structures may have different, untagged record ID structures. 3) It is necessary to use several differing (in structure) DRs in one file. An accompanying DDR must be available for each structure DR.

These can be accomodated by:
1) In enclosing data sets with inline DD and no tags, use the ASCM core and extension, Entry Map, Directory, and Inline Structure, followed by the old data set verbatim. Inter-field terminators are not used, requiring that the fields be in a pre-defined order and all be present or demarkated with the unit separator 1/15.

2) Construct new data sets per 8211, with DDRs and DRs, and data fields with terminators. Include a new record ID field preceding the data fields in the DR, and so indicate in the DDR by the tag 0...1. This fulfills the required format for the DDR. Inter-field terminators are used in the DRs in accordance with the 8211 structure. Note that this causes each data field to be one byte longer than it would otherwise be. For this reason, recovery of the original file verbatim requires that the terminators be removed.

3) It will often be necessary to enclose several 8211-style DDR/DR sets within one transmission. The Data Definition Indicator (DDI) includes information to indicate these groups of DDR/DR/DR... Within each set, the data records will have constant format.

It is to be recognized that the full 8211 structure is more robust than the in-line or otherwise shortened versions, and is therefore to be preferred for new data sets.

MODIFICATIONS BEING CONSIDERED

1. Bring the core extension under ECC. This requires that the separator in the Inline Structure Definitions be changed from an ASCII Unit Separator to the DDI Indicator symbol to avoid a conflict in definitions.

The TAGLIST field, if used, shall contain all tags of both the core and extension.

ACKNOWLEDGEMENTS

The ACSM Digital Cartographic Data Standards Committee is acting under a grant from the US Geological Survey. Participation by the author on the Committee and the preparation of this paper is funded by NASA under a contract to the California Institute of Technology.

REFERENCES

1. Reports of the ACSM DCDS are available from Prof. Harold Moellering (Committee Chariman), Numerical Cartography Lab, Ohio State University, Colombus, Ohio.

Fred Billingsley is a Technical Staff member in the Image Processing Applications and Developmemt Section, Jet Propulsion Laboratory, California Institute of Technology, Pasadena, California, where he was responsible for building the Image Processing Laboratory. More recently, he has been supervisor of the Earth Resources Image Processing Group, a multi-disciplinary organization that is responsible for image processing for Landsat investigations. After a tour of duty at NASA Headquarters he returned to JPL, where he is now concerned with special tasks in data systems for advanced earth observations sensors. He has also been active on many NASA remote sensing working groups and committees. Mr. Billingsley is the author of many publications in the field of digital image processing and is the editor of the chapter on digital image processing in the Revised ASP Manual of Remote Sensing.