

Electronic transfer of geographic information (NTF) —

Part 2: Specification for implementing plain NTF

Committees responsible for this British Standard

The preparation of this British Standard was entrusted by the Information Systems Technology Standards Policy Committee (IST/-) to Technical Committee IST/36, upon which the following bodies were represented:

- Advanced Technology Marine Systems
- Association for Geographic Information
- B Ae Sema Ltd.
- Berkshire County Council
- British Telecommunications plc
- CAM Ltd.
- Cambridgeshire County Council
- Digital Equipment Co Ltd.
- Institute of Terrestrial Ecology
- Laser Scan Ltd.
- Local Government Management Board
- Logica UK Ltd.
- London Fire and Civil Defence Authority
- Ministry of Defence
- Ordnance Survey
- Royal Engineers Survey
- Royal Institution of Chartered Surveyors
- University of Edinburgh

This British Standard, having been prepared under the direction of the Information Systems Technology Standards Policy Committee, was published under the authority of the Standards Board and comes into effect on 15 May 1992

© BSI 09-1999

The following BSI references relate to the work on this standard:
Committee reference IST/36
Draft for comment 91/65602 DC

ISBN 0 580 20875 3

Amendments issued since publication

Amd. No.	Date	Comments
8918	January 1996	Indicated by a sideline in the margin

Contents

	Page
Committees responsible	Inside front cover
Foreword	ii
Section 1. General	
1.1 Scope	1
1.2 Normative references	1
1.3 Definitions	1
1.4 Editorial conventions particular to this Part of BS 7567	1
Section 2. Physical file structure	
2.1 Physical media	2
2.2 Volume structure	2
2.3 Volume termination record	2
2.4 Physical block size	2
2.5 Physical record size	2
Section 3. Physical record structure of plain NTF	
3.1 Records	3
3.2 Fields	3
3.3 Example	4
Section 4. Record implementation	
4.1 Additional records	5
4.2 Records common to all levels	6
4.3 Records for levels 1 and 2	11
4.4 Records for level 3	15
4.5 Records for level 4	21
4.6 Records for level 5	24
Annex A (informative) Summary list of record descriptors	27
Annex B (informative) Example of level 2 data transfer	28
List of references	Inside back cover

Foreword

This Part of BS 7567 has been prepared under the direction of the Information Systems Technology Standards Policy Committee.

BS 7567-2:1992 specifies how NTF, a data transfer standard for a wide range of geographic data, is encoded using plain NTF which is a set of formatting instructions designed specifically for this type of data.

BS 7567-3:1992 may be used as an alternative to this Part by agreement between the donor and recipient.

It is essential for users of this Part to refer to Part 1 of BS 7567.

In section 4, various subclauses are introduced but not used: this device is for the convenience of cross-referral between the implementation specifications for NTF levels 3 and 4. For example, a record for level 3 implemented in 4.4.6 is implemented for level 4 at 4.5.6.

Section 4 in this Part corresponds exactly to section 4 of BS 7567-1:1992. For example, the record implementation specified at 4.4.6 of this Part has its syntax specified in 4.4.6 of Part 1.

A British Standard does not purport to include all the necessary provisions of a contract. Users of British Standards are responsible for their correct application.

Compliance with a British Standard does not of itself confer immunity from legal obligations.

Summary of pages

This document comprises a front cover, an inside front cover, pages i and ii, pages 1 to 40, an inside back cover and a back cover.

This standard has been updated (see copyright date) and may have had amendments incorporated. This will be indicated in the amendment table on the inside front cover.

Section 1. General

1.1 Scope

This Part of BS 7567 specifies the procedures for the formatting of NTF Data Models, the interchange file structures and the detailed record interpretations specified in Part 1 of BS 7567:1992 in accordance with detailed instructions fully specified herein.

NOTE Syntax is also specified for a minority of records which are implemented differently when plain NTF is used instead of BS 6690.

1.2 Normative references

This Part of BS 7567 incorporates, by reference, provisions from specific editions of other publications. These normative references are cited at the appropriate points in the text and the publications are listed on the inside of the back cover. Subsequent amendments to, or revisions of, any of these publications apply to this Part of BS 7567 only when incorporated in it by updating or revision.

1.3 Definitions

For the purposes of this Part of BS 7567, the definitions given in 1.3 of BS 7567-1:1992 apply together with the following.

1.3.1

plain NTF

the custom-designed formatting instructions for NTF specified in BS 7567-2:1992

1.4 Editorial conventions particular to this Part of BS 7567

1.4.1 For the purposes of BS 7567-2:1992, the editorial conventions in 1.4 of BS 7567-1:1992 shall apply.

1.4.2 Use of □

This character is used to signify a space in examples of NTF data.

Section 2. Physical file structure

2.1 Physical media

2.1.1 ½ inch tape

Unless the donor and recipient agree to an alternative medium, the transfer shall be made on unformatted ½ inch 9-track tape with the 8-bit data written at a density of 1600 bpi. Header and trailer records (other than those internal to plain NTF) shall not be used, that is, the tape is unlabelled. The final block on each tape shall be followed by a single tape mark.

2.1.2 Formatted media

If donor and recipient agree to use formatted media, the NTF file shall be written to (or stored on) logically formatted media such as hard disk, floppy disk or logically formatted tape (for example, a ¼ inch cartridge written using the UNIX **tar** command), then the following shall also apply:

- a) The end of record character **{EOR}** shall not be one that is commonly used for carriage control;

NOTE Any control character is liable to special interpretation. It is recommended that the **{EOR}** character be a printing character that does not normally appear in data text strings or NTF control records. For example, a %.

- b) Each physical record shall be followed by the carriage control character(s) for the operating system the NTF file is written under. Thus, on a UNIX system each physical record shall end:

... **{CONT_REC}** **{EOR}** <LF>

and on an MS/DOS system the physical record shall end:

... **{CONT_REC}** **{EOR}** <CR><LF>

where <CR> represents “carriage-return” and <LF> represents “line-feed”.

- c) NTF disk files that are not generated for direct transfer to raw tape shall not have any logical blocks within them.

NOTE File transfer between systems may cause expansion/contraction of carriage control characters. This changes the number of characters in the block and upsets any block reading mechanism.

2.2 Volume structure

If the size of the transfer set exceeds the size of the physical volume, it shall be extended to additional volumes.

A volume shall start with an NTF volume header record and terminate with an NTF volume termination record. When writing data, the donor shall ensure that there is enough space to complete a logical record on the current volume.

NOTE 1 A volume is normally associated with one unit of dismountable physical media. For example, a magnetic tape.

NOTE 2 The existence of more than one volume has no influence on the organization of the data, except that a logical record cannot be continued across volumes.

2.3 Volume termination record

This record shall end both single volumes and complete transfer sets. A field shall indicate continuation or completion of the transfer set.

2.4 Physical block size

Unless the donor and recipient agree to an alternative the block size shall be 2000 characters for transfers on unformatted media.

2.5 Physical record size

Unless the donor and recipient agree to use a fixed length of 80 characters, physical records shall be of variable length with a maximum 80 characters. If fixed length records are used, the donor shall pack as many data fields as possible into the record. Continuation records shall be used when a logical record exceeds the maximum length of a physical record.

NOTE 1 Variable length records are created by the donor and not by the tape facilities of a particular computer system.

NOTE 2 Records are identified within a fixed length block of 2000 characters by an end of record character.

Section 3. Physical record structure of plain NTF

3.1 Records

{REC_DESC}

3.1.1 Record descriptor

All records (except a null record, see 3.1.6) shall start with a specified two character field (format **A2**) which indicates the purpose of the record. The characters are a code matched to the name of the record specified in section 4 of BS 7567-1:1992. Records shall have the format specified at a particular NTF level associated with this descriptor or one defined in a data dictionary in the case of level 5. A correspondence table for record descriptors and record names is provided at Annex A.

NOTE All default record descriptors can be interpreted as two digit numbers.

3.1.2 Continuation mark

{CONT_MARK}

This is an obligatory one-character field in all records except volume terminator and null records, and shall occur immediately in front of {EOR} in any physical record. It shall be **0** (no continuation) or **1** (logical record continues). (See also 3.1.4)

3.1.3 Record terminator

{EOR}

The last character in a record (variable and fixed) shall always be the record terminator, “end of record”. This character shall be defined in the volume header record. The default value shall be <CR> (carriage return) for unformatted media and % (percent) for formatted media. (See also 2.1.)

3.1.4 Continuation record

Where a logical record exceeds 80 characters, the record shall extend into continuation records in the next physical record. In such a case, the continuation mark in the preceding physical record shall be set to **1**.

3.1.5 Splitting records

Where NTF records are written to unformatted media, these shall not be split across blocks, though a continuation record may start in the next block. Null records, consisting entirely of SPCS shall be used to fill blocks. These may be of any length.

NOTE If the penultimate character position in a block is a record terminator, then the next character will also be a record terminator.

3.1.6 Null records (block filling)

Where NTF fields are written to unformatted media, these shall not be split across physical records and blocks and therefore block filling is required when variable length physical records are written to unformatted media. A null record comprising SPC characters shall be used to fill the remaining block space after the last meaningful field. Several such records may be used, each one terminating with {EOR}.

NOTE 1 The first appearance of a null record in a block effectively means that the block contains no more data.

NOTE 2 When the fixed length record option is being used, the last block on a tape will probably require null records to complete it.

3.2 Fields

3.2.1 Splitting fields

Fixed length fields shall not be broken when continuation records are being used. Certain variable length text fields may continue into continuation records, and may also cross blocks. These fields may be split at any convenient point.

3.2.2 Field divider

{DIVIDER}

Unless donor and recipient agree to use a different character the divider for variable fields shall be \.

Dividers shall terminate all variable length {TEXT} fields.

NOTE 1 This character is defined in the volume header record and is only used to terminate variable length text fields.

NOTE 2 {FREE_TEXT} fields in such records as comment, database quality, section quality and volume terminator are not terminated by dividers.

3.2.3 Field descriptors

In interpreting the encoding of a “real” field descriptor, $R_n.d$, the decimal point is not output. Thus the real number 2.75, output using the field descriptor **R4, 2**, would appear as **0275**.

3.2.4 Count fields

Plain NTF shall use additional count fields to indicate the number of times a field repeats and the number of characters in a **{TEXT}** field. The value stored in a count field shall always be greater than zero.

NOTE Count fields are named **{NUM_something}**. Their format is defined as *size* where *size* is the size of the field containing the count.

3.3 Example

Examples of plain NTF record and field structure are provided in Annex B.

Section 4. Record implementation

4.1 Additional records

4.1.1 General

This clause specifies extra physical records that are necessary for the implementation of plain NTF and are relevant at all levels:

- 1) [CONTREC], the continuation record
- 2) Null record, the block fill requirements
- 3) [VOLTERM], the volume termination record

4.1.2 Continuation record

[CONTREC]

4.1.2.1 Record specification

Field name	Format	Use	No data	Range
fields	*	c		

Logical records shall not be continued from one section to the next.

4.1.2.2 Description

- a) To transfer further fields when a logical record is longer than a single physical record.
- b) A continuation record may be associated with any record except volume terminator, comment and null, and it assumes a structure appropriate to the logical record.
- c) The character before {EOR} in any physical record (except for the volume terminator and null records) is always a continuation mark {CONT_MARK}. When a record would otherwise be longer than 80 characters a record may be continued by appropriate use of the continuation mark and continuation records.
- d) With the exception of fields which are of variable length, for example, {TEXT}, fields are not broken across records. If there is insufficient space remaining in a physical record to accommodate a field, the record is terminated and the field continued with the (next) continuation record.
- e) A continuation record may contain only the {DIVIDER} character.

4.1.2.3 Record implementation

Field name	Position	Size	Notes
REC_DESC fields	1: 2 3: *	2	00

4.1.2.4 Implementation details

None.

4.1.3 Null record

4.1.3.1 Implementation details

- a) A null record shall be used to fill a block.
- b) When a null record is more than one character long, the first character shall always be SPC. Remaining characters may be any from the ASCII set.
NOTE It is recommended that only SPCs are used.
- c) A null record of one character shall be {EOR}.
- d) The occurrence of a null record effectively terminates a block of data. It shall not be followed by any other record until a new block is started.

4.1.4 Volume termination record

[VOLTERM]

4.1.4.1 Record specification

Field name	Format	Use	No data	Range
FREE_TEXT	A*	c		
CONT_VOL	I1	c		0 or 1

{CONT_VOL}. Value shall be 0 if the volume completes the transfer, 1 if another volume follows.

4.1.4.2 Description

- a) To mark the end of the transfer set, or where there is more than one volume, to mark the end of the volume.
- b) [VOLTERM] is the last record in a volume, although it may be followed by null records to fill a block.
- c) This record may also complete a transfer set. If {CONT_VOL} is set to 1 then another volume follows. If 0 then the transfer set has been completed.
- d) The existence of more than one volume has no influence on the organization of the data, except that a logical record may not be continued across volumes.

4.1.4.3 Notes on fields

{FREE_TEXT}. Always exists, even if it is only one character.

4.1.4.4 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	99
FREE_TEXT	3: *	*	
CONT_VOL	*: *	1	

4.1.4.5 Implementation details

None.

4.2 Records common to all levels

4.2.1 General

This clause specifies the implementation of records which are common to all levels of transfer.

The common records are:

Type	Record
Comment record	Comment
Volume records	Volume header Volume terminator
Database records	Database header Feature classification Code list Database quality
Data dictionary records	Attribute description Attribute combination
Section records	Section header Section quality Data quality

4.2.2 Comment record

[COMMENT]

4.2.2.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	90
FREE_TEXT	3: *	*	

4.2.2.2 Implementation details

- Comment records may appear anywhere in a transfer set subject to the rule that they do not split a logical record. Therefore, they shall not appear immediately after a record with a {CONT_MARK} set to 1.
- They shall not occur before the volume header record or after the volume termination record.
- {CONT_MARK} in a comment record shall always be 0. If a statement in a comment record requires continuation this shall be achieved by following it with another comment record to ensure that all comment records within a transfer set are distinguished from other records.
- Control characters may be used to format text if necessary, but the donor may need to redefine the end of record marker {EOR} in the volume header record.

4.2.3 Volume header record

[VOLHREC]

4.2.3.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	01
DONOR	3: 22	20	
RECIPIENT	23: 42	20	
TRANDATE	43: 50	8	
SERIAL	51: 54	4	
VOLNUM	55: 56	2	
NTFLEVEL	57: 57	1	
NTFVER	58: 61	4	
NTFOR	62: 62	1	
EOR	63: 63	1	see
DIVIDER	64: 64	1	below

4.2.3.2 Implementation details

- [VOLHREC] shall always be the first record in a transfer set.
- If the transfer set is made up of more than one physical volume, (for example, if the data is being transferred on two magnetic tapes), then there shall be a volume header record at the start of each and every physical volume.
- The following additional fields shall be used for this implementation.
 - {NTFOR} shall specify fixed or variable length fields.

F	=	fixed length fields
V	=	variable length fields
 - {EOR} shall specify a character other than that specified (see 3.1.3) as the record terminator. If the default end of record character is being used, this field shall contain SPC.
 - {DIVIDER} shall specify a character used to divide variable-length fields (see 3.2.2). It may be changed by placing another character in this field. This shall not be {EOR} or SPC, or any character likely to occur in a data field. At level 1, the {DIVIDER} field in [VOLHREC] shall be either \ or SPC since the divider is not used.

NOTE The possible use of * in a data dictionary (levels 3, 4 and 5) and the effect of certain characters when printing out a transfer set is a consideration.

4.2.4 Database header record

[DBHREC]

4.2.4.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	02
DBNAME	3: 22	20	
DDNAME	23: 42	20	
DDATE	43: 50	8	
DDBASE	51: 70	20	
DDBDATE	71: 78	8	

Continuation record

FCNAME	3: 22	20	
FCDATE	23: 30	8	
DQNAME	31: 50	20	
DQDATE	51: 58	8	
DATA_MODEL	59: 60	2	

4.2.4.2 Implementation details

Each database in a transfer set shall start with a database header record. By implication, it terminates a previous database in the same transfer set. It may be followed by:

- 1) data dictionary records;
- 2) feature classification records;
- 3) data quality records.

4.2.5 Feature classification record

[FEATCLASS]

4.2.5.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	05
FEAT_CODE	3: 6	4	
CODE_COM	7: 16	10	
STCLASS	17: 36	20	
FEATDES	37: *	*	

4.2.5.2 Implementation details

None.

4.2.6 Code list record

[CODELIST]

4.2.6.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	42 > 0
FIELD_NAME	3: 12	10	
VAL_TYPE	13: 14	2	
FINTER	15: 19	5	
NUM_CODES	20: 22	3	
CODE_VAL	*: *	*	
CODE_DES	*: *	*	

4.2.6.2 Implementation details

None.

4.2.7 Database quality record

[DBQUAL]

4.2.7.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	06
FREE_TEXT	3: *	*	

4.2.7.2 Implementation details

Continuation records shall not be used to extend this record.

4.2.8 Attribute description record

[ATTDESC]

4.2.8.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	40
VAL_TYPE	3: 4	2	
FWIDTH	5: 7	3	see below
FINTER	8: 12	5	
ATT_NAME	13: *	*	

The following fields may be omitted

FDESC	*: *	*	
NO_DATA	*: *	*	
RANGE_MIN	*: *	*	
RANGE_MAX	*: *	*	
UNITS	*: *	2	

4.2.8.2 Implementation details

- {ATT_NAME} shall be terminated by the {DIVIDER} character.
- If the fields after {ATT_NAME} are included, each shall be terminated by the {DIVIDER} character (as defined in the volume header record). This includes the last field.
- {FWIDTH} shall specify the field width or number of characters used to transfer the attribute value. This count shall not include the mnemonic. When the field is a variable width character field (A*) {FWIDTH} shall contain SPC.

4.2.9 Attribute combine record

[ATTCOM]

4.2.9.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	41
VAL_TYPE	3: 4	2	
NUM_FIELD	5: 7	3	> 0
VAL_TYPE	*: *	2	

4.2.9.2 Implementation details

- This record shall not be used with level 1.
- {NUM_FIELD} shall specify a counter for the number of attribute fields, that is, the number of repeating {VAL_TYPE} fields following {NUM_FIELD}.
- {VAL_TYPE}. A repeating list of {VAL_TYPE} fields collected together. Each {VAL_TYPE} shall have been defined in [ATTDESC] or [ATTCOM] or in accompanying documentation.

[SECHREC]

4.2.10 Section header record

4.2.10.1 Record implementation

Field name	Position	Size	Notes
REC_DESC	1: 2	2	07
SECT_REF	3: 12	10	
COORD_TYP	13: 13	1	
STRUC_TYP	14: 14	1	
XYLEN	15: 19	5	
XYUNIT	20: 20	1	extra field
XY_MULT	21: 30	10	
ZLEN	31: 35	5	
Z_UNIT	36: 36	1	
Z_MULT	37: 46	10	
X_ORIG	47: 56	10	extra field
Y_ORIG	57: 66	10	
Z_DATUM	67: 76	10	

Continuation record

XMIN	3: 12	10	
YMIN	13: 22	10	
XMAX	23: 32	10	
YMAX	33: 42	10	
XY_ACC	43: 47	5	
Z_ACC	48: 52	5	
SURV_DATE	53: 60	8	
LAST_AMND	61: 68	8	
COPYRIGHT	69: 76	8	

Continuation record (see note 1 below)

SQNAME	3: 22	20	
SQDATE	23: 30	8	
SCALE	31: 39	9	
GRID_OR_X	40: 49	10	
GRID_OR_Y	50: 59	10	
PROJ_OR_LAT	60: 67	8	
PROJ_OR_LONG	68: 75	8	

Continuation record (see note 2 below)

SPHER_NAME	3: 12	10	
MAJOR_AXIS	13: 22	10	
ECCENTRICITY	23: 30	8	
FLATTENING	31: 35	5	
PROJECTION	36: 45	10	
PARAMETER_1	46: 53	8	
P_TYPE	54: 54	1	
PARAMETER_2	55: 62	8	

NOTE 1 This record may only be omitted if trailing records are omitted.

NOTE 2 This record may only be omitted if there are no additional user-defined fields.