

Guide to

Specifying electronic typographic markup (SETM)

Committees responsible for this British Standard

The preparation of this British Standard was entrusted by the Paper and Printing Standards Policy Committee (PAM/-) to Technical Committee PAM/35, upon which the following bodies were represented:

- British Printing Industries Federation
- Chartered Society of Designers
- Graphic Reproduction Federation
- Her Majesty's Stationery Office
- Imperial College
- Inter-Universities Computing Committee
- National Centre for Information Media and Technology (CIMTECH)
- Pira (the Research Association for the Paper and Board, Printing and Packaging Industries)
- Publishers Association
- Royal Society
- Trades Union Congress
- University of Reading

This British Standard, having been prepared under the direction of the Paper and Printing Standards Policy Committee, was published under the authority of the Standards Board and comes into effect on 31 July 1991

© BSI 1991

The following BSI references relate to the work on this standard:
Committee reference PAM/35
Draft for comment 88/36402 DC

ISBN 0 580 19289 X

Amendments issued since publication

Amd. No.	Date	Text affected

Contents

	Page
Committees responsible	Inside front cover
Foreword	2
Guide	
0 Introduction	3
1 Scope	7
2 Definitions	7
3 General	8
4 Specifying character sets	9
5 Starter set of markup tags	13
6 Specifying markup tags	30
Appendices	
A Guidance on the specification of character sets	36
B Guidance on the specification of new or altered markup tags	37
C Examples of generalized markup tags	38
D Examples of symbol reference tags	41
Tables	
1 Basic set of text characters	12
2 Validity of units of measurement	27
General index	42
Subject index of markup tags in the starter set	44
Index of markup tags in the starter set	46

Foreword

This British Standard has been prepared under the direction of the Paper and Printing Standards Policy Committee and gives guidance on ways of specifying the typographic form of a document that is being prepared electronically for typesetting without the need for rekeying at a later stage. The method is based upon the use of markup techniques, and this guide defines how these markup techniques are to be specified and used.

This standard also provides guidance on specifying the set of characters available on the word processing or other text preparation system being used to prepare the document. The purpose of this method is to enable all characters in the set and their coded representations to be identified prior to typesetting, so that any necessary conversion can be performed correctly and completely prior to typesetting.

Observance of procedures in this guide does not require the use of any particular set of typesetting instructions. It therefore does not guarantee a meaningful interpretation of any particular typographic instruction on any specific occasion.

However, this guide provides a 'starter set' of typographic 'markup tags' (see clause 5) that may be interpreted equally well by people or machines, and a way of specifying additional or replacement markup tags. In most cases it will enable those who already use another set of markup tags to continue to use them.

It is recognized that there are other kinds of markup technique that may be applied to electronic text preparation. The relationship between this British Standard and these other techniques needs to be clear. In particular, the complementary relationship between this British Standard and BS 6868 is defined in 3.3 and further explained in 0.8 and appendix C.

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

Guide

0 Introduction

0.1 General

NOTE 1. It is assumed that those intending to use this British Standard are familiar with the use of electronic text preparation systems such as word processors.

This British Standard guide is concerned only with how to make the best use of an electronic text preparation system in the specific task of preparing text for typesetting. This introduction gives the background to the guide and explains concepts upon which this British Standard is based. The guide includes a technique for preparing electronic text for typesetting that is applicable to word processors, microcomputers and other electronic text preparation systems.

NOTE 2. Appendices A and B contain further guidance on specifying character sets and new markup tags. Unless intending only to use the starter set defined in clause 5, the user will find it helpful to read these as well as the introduction.

0.2 What is 'electronic typographic markup' ?

Markup is a term derived from the practices of designers and editors in preparing written or typed text for typesetting. Traditionally, to 'mark up' text meant to annotate the manuscript with 'marks' that were abbreviated instructions for the operator of the typesetting system to follow, to achieve the correct style and layout of the text.

On electronic systems text is represented by a sequence of signals that are the coded representations of individual text characters. Annotation of electronic text to convey style and layout instructions therefore involves the insertion into the text of short sequences of coded characters whose subsequent interpretation will be as typographic instructions or markup, rather than as text.

Each typographic instruction is coded as a sequence of characters known as a 'markup tag'. The choice of sequence may be a mnemonic abbreviation of the instruction it represents. For example, this guide defines (in clause 5) a markup tag coded as [i] to represent the instruction, 'the following text is to be typeset in italic'. Such a sequence may be entered from the keyboard at the same time as the text or inserted at a later editing stage.

Such encoded markup is known as 'electronic typographic markup' and is the primary concern of this guide.

The following examples are taken from the starter set of typographic markup tags defined in clause 5. Other markup tags may be used with the same effect if so specified using the method defined in clause 6. In each example the text with embedded markup tags is presented first, followed by the typeset result.

(a) Changing type style

In [i]italic [ro]and [b]bold [ro]type

In *italic* and **bold** type

(b) Changing type size

[s 10pt]To make type [s 8pt]smaller

To make type smaller

(c) Indenting text

[ih 1pi 6pt]A paragraph with a hanging indent of 1 pi and 6 pt

A paragraph with a hanging indent of 1 pi and 6 pt

(d) Creating rules

[rh 1pt 18pi]

0.3 Other kinds of electronic markup

Another kind of electronic markup recognized in this guide conveys descriptive information about the logical (as opposed to the typographic) structure of the text. For example, the meaning of a particular markup sequence might be, 'this is the start of a new chapter'. Such markup conveys no directive of how the text is to be processed, and is known as generalized markup.

A third kind of electronic markup recognized in this guide is used to represent special text characters or symbols for which there is no convenient means of entry from the keyboard. This kind of markup is referred to as symbol reference markup.

NOTE. The standard generalized markup language (SGML) specified in BS 6868 defines the use of generalized markup and symbol references ('entity references' in SGML terminology) in ways and for purposes that are beyond the scope of this guide. Their use as defined in this guide implies a simpler and more restricted interpretation in preparing certain kinds of text for typesetting.

0.4 Why a British Standard guide?

Electronic markup techniques have been under development for as long as authors, publishers and designers have been using electronic systems to prepare text for typesetting. There are considerable time and, in many cases, cost advantages to be gained from the judicious use of such techniques, which can save double keyboarding.

It is now widely accepted that the standardization of electronic markup techniques has benefits for all concerned. Lack of standardization necessitates the learning of many differing techniques: a cost that is in the end borne by someone. Standardization, however, can only come about through the co-ordinated efforts of those who stand to benefit.

While no-one can claim that the voluntary adoption of a guide such as this automatically and inevitably leads to tangible benefits for the user, it does cause the odds in favour of such benefits to be shortened considerably.

0.5 What does this British Standard help people to do?

This guide is intended for use by people who are concerned with preparing text for typesetting: authors, designers, publishing house staff and typesetting house staff. It is concerned specifically with the preparation of text for typesetting in circumstances in which typographic instructions need to be included in the text on a terminal that does not control the typesetting process, such as on a production editor's word processor.

The main concern of this guide is the expression of typographic instructions, that is to define how text is to be typeset. It provides a technique for coding these instructions that is independent of both the text preparation system and (as far as possible) the typesetting system and output device (typesetter, for example) by which the text is to be imaged. Its application presupposes that the necessary design decisions have been taken.

0.6 Are there other uses of this British Standard?

This guide can be used for the exchange of text between typesetting systems, by providing a method for representing typesetting instructions. Other uses are possible, such as to provide a basis for generic training in typographic markup techniques for graphic designers and typesetting system operators.

0.7 What does this British Standard not help people to do?

Any finite set of markup tags is necessarily restricted in its application. The starter set (see **0.15** and clause **5**) does not attempt to provide the markup tags required for complex typesetting tasks such as tabulation, displayed mathematical setting and page make-up.

It is presumed that any person or organization intending to use this guide has the desire to prepare text for subsequent typesetting on a system that is 'remote' from the text preparation process. This means that typesetting is a process distinct from preparation, almost certainly on a physically separate system, very probably on a system whose typesetting language is unknown or unspecified. While this may seem to restrict the field of application, a wide range of circumstances is still covered.

There are occasions when text preparation and typesetting take place on one and the same system. This guide is unlikely to find application in such circumstances.

In other circumstances the precise typesetting instructions are determined by an operator interacting with a system that gives a graphic display of the eventual page. Desktop publishing systems and a variety of interactive page layout systems fall into this category. Although text preparation may in such cases be performed as a distinct operation prior to the typesetting process, it is unusual for typesetting instructions to be inserted at the text preparation stage. This therefore represents another field of typesetting activity in which this guide is unlikely to find frequent application.

0.8 What is the relationship of this British Standard to SGML?

This guide is complementary to the standard generalized markup language (SGML), as it is concerned specifically with typography, while SGML makes no specific provision for the inclusion of typographic instructions in text. The application of both this guide and SGML may prove beneficial in some text preparation tasks. Between them they provide the user with a degree of flexibility over the way that text can be subsequently processed that exceeds what has been hitherto practicable except to specific (system-dependent) circumstances.

In other circumstances (for example, when the text is required for purposes in addition to typesetting, such as database creation; or the text may be required to be typeset later to a different design) an application of SGML is likely to be more useful. Authors wishing to use markup techniques may find that, unless they have some knowledge of typography and typesetting, generalized markup will take them as far as they wish to go.

SGML defines a system of markup that offers a wide range of options for describing a document in terms of its logical structure. Although specifying electronic typographic markup (SETM) recognizes generalized markup tags, it does so to a very restricted extent in comparison with SGML. As a consequence, SGML documents will often contain markup that cannot be recognized by systems that implement SETM.

NOTE. For further explanation of this restriction see appendix C.

If text is to remain in machine-readable form for future (re-)use, the user is advised to mark it up in accordance with SGML, as it is generally easier to convert generalized markup for typesetting than to convert typographic markup for other text processing activities.

0.9 Is there an international standard equivalent to this British Standard?

There is no international standard that provides an exact or near equivalent method to that defined in this guide. It should be noted, however, that it is intended that future international standardization should define a language for expressing the relationship between the logical structure of a text document, as expressed using SGML, and its layout (typographic) structure(s).

0.10 How is the guide organized?

The aim of this introduction is to help the reader to understand the three substantive parts of the guide:

- (a) specifying character sets (see clause 4);
- (b) the starter set (see clause 5);
- (c) specifying markup tags (see clause 6).

Clause 4 describes a technique for specifying all the characters to be used in preparing the text: capital and lower-case letters, numerals, punctuation, etc. The method may seem rather formal, with characters to be listed in precise order; this is necessary in order to minimize the risk of the prepared text being misinterpreted at the typesetting stage; there is, after all, no compositor to correct mistakes. Lack of standardization of computer systems is the principal cause of such misinterpretation.

Clause 5 defines a set of markup tags. It is designed to be a large enough set for many straightforward text preparation tasks, and is called the 'starter set'. The starter set may be expanded as described in clause 6.

Clause 6 describes a technique for specifying new markup tags of any of the three categories recognized by this guide: typographic, generalized and symbol reference. It can be used to make modifications or additions to the starter set, or alternatively to define a completely new set of markup tags. The defined method is strictly formal, to facilitate machine interpretation and minimize the risk of ambiguity.

Appendices A to D give further guidance on the use of generalized and symbol reference tags.

There are three indexes: a general index and two indexes to the markup tags of the starter set, the first in alphabetical order of definition, the second in alphabetical order of tag identifier.

0.11 How is this guide to be used?

It is recommended that the user follows the following four stages in order to make best use of this guide.

- (a) *Stage 1.* When the user has become familiar with the guide and its application, it is necessary to establish that the operator in charge of the typesetting system is also familiar with this guide and that the typesetting system can accept text coded with standard typographic markup.

- (b) *Stage 2.* It is necessary to determine whether to supply the typesetter with a specification of the characters available on the text preparation system, in which case the method of clause 4 should be followed. This is necessary if any of the following are new or have changed:

- (1) the typesetting organization;
- (2) the typesetting system;
- (3) the text preparation system, e.g. word processor;
- (4) the characters to be used;
- (5) the coded representation of the characters to be used.

- (c) *Stage 3.* The author, or whoever originates the typographic markup in electronic form, determines whether the starter set of markup tags of clause 5 provides all the typographic markup tags needed to prepare the text in question for typesetting. If so, reference to the starter set is all that is now needed to proceed with preparing the text.

- (d) *Stage 4.* If the starter set is not adequate, determine whether it is necessary to supply the typesetter with a specification of the additional or modified markup tags required, in which case the method described in clause 6 should be followed. This will be necessary if the set of markup tags to be used deviates from the set previously specified and therefore already known to the typesetter.

0.12 Using this British Standard to solve problems in text preparation

NOTE. It is suggested that the remainder of clause 0 should only be read after some familiarity with the methods defined has been gained by reading the body of the guide.

0.12.1 General

Typographic markup is an expression of how the designer intends text to be typeset. The principal problem encountered in preparing the text for typesetting is how to express the text content and the intended design clearly and unambiguously, given the limited facilities available on most text preparation systems.

0.12.2 Character sets

One of the limitations of text preparation systems is the limited range of graphic characters, falling far short of the user's needs in many situations. This limitation manifests itself in two ways:

- (a) ambiguity in the meaning of characters;
- (b) absence of characters altogether.

Clause 4 describes a technique for specifying the character set of a text preparation system as the user intends it to be interpreted.

The method divides characters into graphic and control characters, the graphic characters being further subdivided into a 'basic set' and special symbols.

It is strongly recommended that every character in the basic set should have only one interpretation. For example, the user may be tempted to use the same character ‘-’ to mean both ‘hyphen’ and ‘minus’. Ambiguity in the meaning of any character can cause serious problems at later stages in typesetting. An example is considered in appendix A.

This guide recognizes an alternative means of representing special symbols using symbol reference tags. Each symbol is represented not by a single graphic character, but by a markup tag that means: ‘at this point place the symbol ...’ An example is given in appendix B.

0.12.3 *Typographic markup tags*

Having defined how the text content is to be included unambiguously, there remains the problem of how to include instructions as to the intended design.

Traditional copy markup techniques provided a range of signs and abbreviations that enabled the designer to convey to the compositor an armoury of typographic instructions on how the text should be typeset. It has to be said that this armoury was not always reliable, as a fair amount of misinterpretation of ambiguous marks has been a constant factor in the introduction of errors into the typesetting process. Mechanical processes such as typesetting cannot interpret ambiguous instructions, so it is of paramount importance that ambiguity is avoided in typographic markup.

Potential users of this guide may have little or no experience of the use of coded typographic markup to convey typesetting instructions. For this reason a starter set of typographic markup tags that is extensive enough for simple typesetting tasks is described in clause 5.

More complex typesetting tasks may require a more complex set of typographic markup tags than the starter set, and may additionally require the use of generalized markup and symbol reference tags. For this reason a technique for specifying additional markup tags is given in clause 6.

NOTE 1. Of all complex typesetting tasks, pagination is the least likely to benefit from the use of markup, as it can only take place once text has already been arranged into lines. Only loose control can be exercised over page endings until the text has been through the typesetting process.

NOTE 2. The ASPIC Handbook contains an existing set of markup tags that may help those wishing to extend the use of typographic markup into more complex tasks such as tabulation.

0.13 Problems encountered in conversion for typesetting

Most typesetting systems have very limited capabilities for interpreting text that has been prepared on other systems. Many typesetting systems can now be connected to devices that perform media conversion, i.e. the interpretation of disks and tapes from one format into another, and code conversion. More sophisticated code conversion systems are capable of interpreting

coded text according to contextual cues and patterns, making it possible to take text without any coded markup and add markup ‘intelligently’ without manual intervention.

Nevertheless, any method for conveying typographic instructions to a typesetting system in the form of coded markup tags is, at best, only a partial solution to the problem. Certainly such a method will work only if the set of markup tags to be used is intelligently designed and unambiguously defined.

0.14 Character set specification

The technique for specifying the character set of a text preparation system, described in clause 4, involves the creation of a series of formal ‘statements’ designed to be interpreted by a computer system or its operator.

Each statement consists of a line of text containing characters in a predetermined sequence. It is by knowing what comes next in the sequence that the computer system or operator is able to work out the relationships between character and coded representation. For this reason it is vital that the correct sequence is always followed.

As only a proportion of users will be concerned directly with character set specification, a detailed discussion of the method is not included here, but is to be found in appendix A.

0.15 Starter set

The starter set is a set of typographic markup tags designed to be used by those who do not already have a preferred set of markup tags and would like the reassurance of using a set that is acceptable to as wide as possible a range of typesetters.

The starter set does not aim to cater for more than straightforward typesetting needs. It does not provide markup for tabular matter, mathematical or chemical formulae, text block handling, single- or multi-column page makeup, all of which involve the use of instructions that are outside the scope of many text preparation systems.

The starter set adopts the convention of using square brackets [and] to indicate the start and end of a sequence of characters forming a markup tag. Such pairs of characters are referred to as delimiters. The starter set also uses the solidus / to indicate termination or cancellation of typographic instructions.

NOTE. The need to use square brackets as text characters, or their absence from the character set of a partial text preparation system, will necessitate a change of delimiters using the method described in clause 6. This is further explained in appendix B.

0.16 Markup tag specification

The technique for specifying markup tags, described in clause 6, involves the creation of a series of formal ‘statements’ designed to be interpreted by a computer system or its operator. In this respect there are similarities between this method and that adopted for the specification of character sets.

The principal difference between the method of clause 4 and that of clause 6 is that in the latter the statements defining each individual markup tag normally run to several lines of text. Yet they are highly formalized, again to avoid possible misinterpretation by the computer program, programmer or system operator.

Markup tags are defined to be of one of three kinds: typographic, generalized or symbol reference. Each is defined in terms of delimiters, identifier and any fields for variable information within the tag, such as measurement values or identifying names.

If it is the user's intention to stick to the starter set, clause 6 can be ignored. If the user wishes to change the set of markup tags available in any way, even simply a change of delimiters, a markup tag specification should be prepared.

Further assistance in preparing a markup tag specification can be found in appendix B.

1 Scope

This British Standard provides guidance on specifying typographic forms of a document that is to be prepared electronically for typesetting.

This guide applies to text prepared for typesetting on word processors and other text preparation equipment.

Clause 4 describes how to specify the character set of the text preparation equipment.

Clause 5 describes a set of typographic markup tags, called the starter set.

Clause 6 describes how to specify markup tags in addition to or as replacements for those described in clause 5.

Appendix A gives guidance on how to specify the character set of the text preparation equipment, in support of the application of clause 4.

Appendix B gives guidance on how to specify markup tags, in support of the application of clause 6.

It is assumed that the markup tags used in any specific situation in the preparation of text for typesetting have meanings that enable them to be interpreted by the typesetting system selected to receive the text so prepared.

NOTE. The titles of the publications referred to in this standard are listed on the inside back cover.

2 Definitions

For the purposes of this British Standard, the following definitions apply.

2.1 argument

A character sequence that forms the qualifying or quantifying variable part of a markup tag.

2.2 available measure

The maximum length of line that is allowed when dividing text into lines, taking into account any indentation.

NOTE. See 2.10.

2.3 delimiter

A character sequence that marks the start or end of a markup tag.

2.4 field

A single (quantitative or qualitative) constituent of an argument.

2.5 fount (synonym font)

A set of graphic characters and associated imaging instructions, often related by being of a single typeface (see note 2 of 5.4.2.1).

2.6 formatter

A set of computer programs that determines the arrangement of text on an imaging surface (paper or screen) according to either typographic instructions specific to the text or standing instructions on character positioning, line and page ending, etc.

2.7 generalized

Descriptive of a markup tag whose meaning is not specific to a procedure.

2.8 identifier

A character sequence that identifies the role of the markup tag of which it is part.

2.9 markup tag

A character sequence used to convey typographic instructions or other information about the intended structure, content or mode of presentation of text, or other procedural attributes.

2.10 measure

The maximum length of line that is allowed when dividing text into lines in the absence of any indentation.

2.11 parser

A programmed set of instructions designed to recognize the structure of data.

NOTE. An SGML parser recognizes the logical structure of text data as defined within a document type definition (see BS 6868).

2.12 separator

A character sequence that marks the start or end of a field, and may convey information on how to interpret the field that precedes or follows it.

2.13 symbol reference

Descriptive of a tag whose meaning is to mark the location of a symbol that, by necessity or for convenience, has not been explicitly included in the text at that point, but is to be included on final presentation.

2.14 text preparation system

Any computer-assisted system for the preparation of textual documents, including word processing systems, typesetting systems and other computer systems equipped with text editing, word processing or typesetting programs.

3 General

3.1 Organization of specification data and marked-up text

This British Standard, apart from providing a starter set of typographic markup tags, provides methods for specifying character sets and markup tags. These methods involve the creation of information that can be interpreted by a receiving typesetting system.

Where this guide is used any of the following three distinct classes of textual data should be conveyed:

- (a) a formal specification of the character set of the text preparation system (see clause 4);
- (b) a formal specification of those markup tags to be used in text preparation that are in any way different from the markup tags of the starter set (see clauses 5 and 6);
- (c) the text prepared for typesetting, using embedded markup tags in accordance with this guide.

For the receiving typesetting system to make sense of the information supplied it is necessary that the information be organized properly. The three classes of data identified in (a), (b) and (c) should therefore be conveyed to the receiving typesetting system in the following sequence:

- (1) character set specification;
- (2) markup tag specification;
- (3) marked-up text to be typeset.

The character set specification and the markup tag specification should each be wholly contained within a single data file, and may both be contained in the same file. In any case, it is essential that the retrieval of the specifications and marked-up text in the above sequence is possible.

3.2 Hyphenation and justification

The presence of hyphens generated manually or by a hyphenation process on a text preparation system would create ambiguity, as a typesetting system would not be able to distinguish between hyphens that should always be typeset ('hard' hyphens) and those that should only be typeset if dividing a word between one line of text and the next ('soft' hyphens). Therefore 'soft' hyphenation, whether inserted manually or by a hyphenation process, should not be used in preparing text for typesetting in accordance with this guide.

Text for typesetting may be prepared in justified form, but this is not recommended. Justification of text may cause serious problems in conversion for typesetting, as many text preparation systems use methods of justification which involve altering the coded representation of characters, for example, space characters and characters at the ends of words.

3.3 Interworking with SGML

It is expected that this guide will find application in circumstances in which other British Standards are applied. The interworking of this guide with SGML is based upon a recognition that SGML defines structure but does not define how typographic information is to be included in a document (although it provides a variety of facilities for doing so).

Typographic markup tags, defined and used in accordance with this guide, may be included in an SGML document in one of the following ways.

- (a) The typographic information can be associated with logical elements of a document by the definition of one or more layout attributes for each element; such information would then be included in a document by the inclusion of appropriate attribute names and values within generalized markup tags (see 6.7.5.8).
- (b) The typographic information can be included within marked sections which may be either ignored or treated as character data by an SGML parser.
- (c) The typographic information can be included as processing instructions which may or may not be placed within marked sections as above, to be either ignored or passed to a typographic markup processor, e.g. a formatter, by an SGML parser.
- (d) The typographic information can be included as character data which is treated as text by an SGML parser.

NOTE. Alternatively, typographic markup tags may be used as a basis for specifying the layout of an SGML document by the creation of an electronic style sheet, associating logical elements with their intended mode of presentation. This may be achieved at present using 'concurrent structures' (a feature of SGML) to link layout and logical structures. In future it is intended that there should be a standard method for associating logical structure with intended layout, using the document style semantics and specification language (DSSSL): a proposed international standard that is currently at the stage of draft.

The choice of how to include typographic markup tags within an SGML document depends upon the specific circumstances of the individual user.

4 Specifying character sets

NOTE. Reference should be made to appendix A.

4.1 General

The technique for specifying the character set of a text preparation system should follow the model given in 4.3.2.

The method enables the creation of a character set specification that may be conveyed between the text preparation system on which text is originated in electronic form and a system receiving the text for the purposes of typesetting.

A character set specification specifies how individual text characters and spaces are represented, and how characters that control the display of text on the text presentation system (for example, end-of-line characters) are represented.

4.2 The structure of a character set specification

The method should result in information being presented to the receiving system in the following order:

- specification of how a basic set of text characters is represented;
- specification of how other graphic characters are represented;
- specification of how space (for example, between words) is represented;
- specification of how characters that control the presentation of text are represented.

NOTE. The complete character set specification precedes the markup tag specification and the marked-up text (see 3.1).

4.3 The basic set of text characters

4.3.1 General

The basic set of text characters is those graphic characters of the text preparation system that are defined in table 1 and appear in the model specification given in 4.3.2.

The basic set specification should consist of three lines of text, each ended by the use of the end-of-line control function on the text preparation system, i.e. NEW LINE, RETURN, ENTER or JUSTIFY, according to keyboard layout convention. The end-of-line control function should not be inserted elsewhere in the basic set specification.

Each line of the basic set specification should contain seven groups of four text characters, in the precise sequence shown in the model specification below. Each group of four characters should be separated by a single space character, to be inserted by a single depression of the space-key or space-bar on the keyboard of the text preparation system. The space character should not be inserted at the beginning or end of a line, or elsewhere in the basic set specification.

The first two characters in the first group on the first line of the basic set specification should be the same character, the question mark ?. The two question marks indicate the start of the specification.

If any text character defined in table 1 is not available on the text preparation system, this is indicated in the basic set specification by entering a question mark ? in the precise position in the sequence where the character would have appeared had it been available.

4.3.2 Model specification of the basic set

The text characters that form the basic set should be given in the precise sequence indicated by the following model specification.

```
??AB CDEF GHIJ KLMN OPQR STUV WXYZ
&'ab cdef ghij klmn opqr stuv wxyz
()01 2345 6789 * + , - . / : ; < = > ? ! % [ ]
```

4.3.3 Example of specifications of the basic set

If all text characters in the basic set defined in table 1 are present on a text preparation system, the specification of the basic set should be identical to the model specification given in 4.3.2.

A basic set that does not contain the characters * and % may be given as follows:

```
??AB CDEF GHIJ KLMN OPQR STUV WXYZ
&'ab cdef ghij klmn opqr stuv wxyz
()01 2345 6789 ? + , - . / : ; < = > ? ! % [ ]
```

4.3.4 Interpretation of ambiguities

Only the question mark ? should appear more than once in a basic set specification. If another graphic character appears more than once, the receiving typesetting system should interpret the character according to its last occurrence in the specification and interpret its previous occurrences as if the question mark had occurred in these positions.

For example, the following specification

```
??AB CDEF GHIJ KLMN OPQR STUV WXYZ
% 'ab cdef ghij klmn opqr stuv wxyz
()01 2345 6789 * + , - . / : ; < = > ? ! % [ ]
```

in which the per cent sign % appears twice is equivalent to the specification

```
??AB CDEF GHIJ KLMN OPQR STUV WXYZ
? 'ab cdef ghij klmn opqr stuv wxyz
()01 2345 6789 * + , - . / : ; < = > ? ! % [ ]
```

in which the ampersand & is interpreted to be not available on the text preparation system.

4.3.5 Meaning of characters

The meaning or role of each character in the basic set should correspond to the definition of the character given in table 1.

Characters with different meanings or roles from those defined in table 1 should be specified as 'other graphic characters' using the method described in 4.4.

For example, the character appearing at the beginning of the second line of the basic set specification may be either an ampersand & if this is available or a question mark ? if it is not. The receiving typesetting system should be entitled to interpret the meaning of each character according to its position in the basic set specification. For example the following specifies that the ampersand & is available in the basic set but appears as %, while the per cent sign is not available.

```
??AB CDEF GHIJ KLMN OPQR STUV WXYZ
%'ab cdef ghij klmn opqr stuv wxyz
()01 2345 6789 * +, - ./:; < = > ? !%[ ]
```

As another example, braces {} are not part of the basic set, but it is possible to use them in the role of square brackets by inserting them in the last two positions in the basic set specification, as follows:

```
??AB CDEF GHIJ KLMN OPQR STUV WXYZ
&'ab cdef ghij klmn opqr stuv wxyz
()01 2345 6789 * +, - ./:; < = > ? !%{ }
```

However, it should be remembered that the receiving typesetting system will in this case interpret the braces to be square brackets.

4.3.6 Founts and the specified character set

It is assumed that the user of the text preparation system is free to use any character of the basic set and any special character with any selection of typeface (see 5.4.2.1) and that this can be catered for by the typesetter. For the purposes of text preparation a 'fount' should therefore contain all specified graphic characters.

NOTE. Founts on typesetting systems vary in their character repertoire. The founts of the receiving typesetting system will usually vary from the founts assumed available during text preparation. The user of the text preparation system can under such circumstances at best expect a near match for every character used in preparing the text. For example, it is extremely unlikely that a typesetter will be able to provide an italic version of an equals sign, in which case the roman version will probably appear instead.

4.4 Other graphic characters

4.4.1 General

After specifying the basic set other graphic characters may be specified.

The specification of other graphic characters should consist of a series of lines of text. Each line should contain the specification of a single graphic

character. Each line should be ended by use of the end-of-line control function on the text preparation system (see 4.3.1).

Each individual character specification should contain the character as keyed on the text preparation system followed by a description of the meaning of the character, in accordance with the following syntax:

- (a) character as keyed;
- (b) one or more space characters;
- (c) two colon characters;
- (d) character description;
- (e) one colon character.

The character description should contain text composed only from the graphic characters of the basic set, including the space character as necessary.

4.4.2 Example of character specification

The following is an example of how to specify a character that is not in the basic set.

\$: : dollar sign :

4.4.3 Characters to which this method applies

The method for specifying graphic characters other than those in the basic set should apply except in the case of symbols that are keyed using symbol reference tags specified using the method given in clause 6.

NOTE 1. In practice, the method is likely to be applied to characters that can be keyed directly on the text preparation system, or to those for which there is a sensible substitute character, e.g. * specified to mean a multiplication sign. It is symbols and logos for which there is no sensible single-key convention on the text preparation system that are likely to be entered using symbol reference tags.

The method may be used to specify characters that are created by a combination of keystrokes on the text preparation system. The keying of characters that involves the use of the control functions such as BACKSPACE is not recommended.

NOTE 2. Graphic characters that may be confused with characters in the basic set should be carefully specified using this method. Other characters that are commonly but not uniformly available on text preparation systems should also be specified using this method. The following are included in these two categories:

- (a) quotation marks: single, double, left, right;
- (b) accents and accented letters;
- (c) typographic rules, such as em and en rules;
- (d) typographic spaces, such as em space;
- (e) left and right braces {curly brackets};
- (f) minus sign;
- (g) low line (underline) _;
- (h) modulus sign |;
- (i) reverse solidus \;
- (j) multiplication sign;
- (k) commercial 'at' sign @;
- (l) pound sterling sign £, dollar sign \$;
- (m) number (hash) sign #.

4.5 Space characters

The basic set of characters (see 4.3.1) contains the normal space character used in text preparation. In addition, text preparation systems may produce other kinds of space (excluding 'tabs', see 4.6.2.2) as follows:

- (a) a no-break space character, used to indicate that the line may not end at that point;
- (b) typographic spaces, used to create spaces of a specified width (relative to the current type size);
- (c) justification spaces, used on some text preparation systems to indicate that the width of this space is to be determined so as to justify the line.

No-break space characters and typographic spaces are generated by specific keyboard action, normally resulting in coded representations that can be readily recognized by the receiving typesetting system. These kinds of space may be specified using the method given in 4.4; or they may be entered into the text using symbol reference tags specified using the method given in clause 6.

Justification spaces are generated by the text preparation system in place of normal spaces following some instruction to the system to justify text. The coded representation of a justification space may not be easy to recognize, and it is not possible to guarantee their correct interpretation by the receiving typesetting system. The use of justification spaces, and therefore of justification in general, on the text preparation system is not recommended. The technique for specifying new line functions (see 4.6.1) may be used to attempt to specify justification spaces, if their use is strongly desired for some reason.

NOTE. It is not necessary to use justification on the text preparation system in order to obtain this arrangement of text when typeset. Justification of text can be specified using a typographic markup tag, such as one of those given in the starter set (see clause 5).

4.6 Presentation control functions

4.6.1 End-of-line control functions

4.6.1.1 General

Text preparation systems allow text to be divided into lines either by instruction from the keyboard (NEWLINE, RETURN, ENTER, etc.) or under the control of the text preparation program each time text fills the available measure (space between the margins).

The manually-keyed end-of-line control function is specified by its use in the basic character set specification (see 4.3.1).

The program-generated end-of-line should be specified by creating two lines of text in which the end of the first line is program-generated and the end of the second line is manually-keyed.

4.6.1.2 Justification not to be used

If justified text is not to be used on the text preparation system, the two lines of text should

contain the word TEXT repeated as many times as is necessary to force the text preparation system to generate an end-of-line.

A single normal space character should be keyed following each word TEXT except the last, which should be followed by the manually-keyed end-of-line function.

There should be at least one occurrence of the word TEXT on the second line of the specification. Following the creation of the two lines of text as just described, a question mark should be inserted immediately in front of the first word TEXT on the second line.

The completed specification should correspond in pattern (but not necessarily in number of words) to the following model specification:

```
TEXT TEXT TEXT TEXT TEXT TEXT TEXT
?TEXT
```

4.6.1.3 Justification to be used

NOTE 1. It is not necessary to use justification on the text preparation system in order to obtain this arrangement of text when typeset. Justification of text can be specified using a typographic markup tag, such as one of those given in the starter set (see clause 5).

If justification is to be used on the text preparation system, it is necessary to use a modified method for specifying both the program-generated end-of-line control function and the justification space.

The two lines of text should consist of the word JUSTIFIED followed by the word TEXT repeated as many times as is necessary to force the text preparation system to generate an end-of-line. In other respects the specification is to be created as in 4.6.1.2, as follows:

```
JUSTIFIED TEXT TEXT TEXT TEXT TEXT TEXT
?TEXT
```

NOTE 2. On some text preparation systems the justification space is only distinguishable from a normal space if the width is different. This is, of course, mostly the case, but it is just possible that a version of the above specification could contain a first line which happens to justify with normal spaces between words. The user should take care to avoid this possibility, such as by adjusting the measure (margins) to ensure that some spaces between words will have to be wider than the normal space.

If a mixture of justified and unjustified text is to be included, a second specification should follow that for justified text, replacing the word JUSTIFIED with the word UNJUSTIFIED, as follows:

```
UNJUSTIFIED TEXT TEXT TEXT TEXT TEXT TEXT
?TEXT
```

NOTE 3. The user should take care to ensure that the specification for justified text is keyed only after an instruction to the text preparation system to justify text has been given in the way demanded by that system. Similarly, keying the specification for unjustified text should follow an instruction to the text preparation system to set text unjustified.

4.6.2 Other presentation control functions

4.6.2.1 General

Other presentation control functions as follows may be specified following the program-generated end-of-line control function:

- (a) functions influencing the position of following text;
- (b) functions influencing the appearance of following text.

4.6.2.2 Position control functions

A position control function determines the position of following text relative to preceding text.

NOTE 1. End-of-line control functions are of this kind, but have already been specified.

A position control function should be specified by text that has the following sequence:

- (a) the word BEFORE;
- (b) the position control function;
- (c) the word AFTER;
- (d) one or more normal space characters;
- (e) two colons;
- (f) a description of the function;
- (g) one colon;
- (h) manually-keyed end-of-line.

The description should be composed entirely of characters from the basic set of text characters, including the space character.

For example, the specification of a tab function would have the following appearance:

BEFORE AFTER :: tab function :

NOTE 2. Other control functions of this kind may include functions to raise or lower characters relative to the line of text. However, functions that affect both position and appearance, such as functions for creating superscript and subscript characters, should be specified using appearance control functions (see 4.6.2.3).

4.6.2.3 Appearance control functions

An appearance control function determines the appearance of following text. Usually, either such functions exist in on-off pairs or the same function both switches on and switches off the effect according to the current state.

An appearance control function should be specified by text that has the following sequence:

- (a) the word OFF, followed by one space;
- (b) the control function that causes the change of appearance;
- (c) the word ON, followed by one space;
- (d) the control function that cancels the change of appearance;
- (e) the word OFF;
- (f) one or more space characters;
- (g) two colons;
- (h) a description of the control function;
- (i) one colon;
- (j) manually-keyed end-of-line.

The description should be composed entirely of characters from the basic set of text characters, including the space character.
For example a control function to make text bold would be specified as follows:

OFF ON OFF :: bold function :

NOTE. It is not necessary to use appearance control functions to control changes of style at the typesetting stage. Explicit typographic markup tags, such as those given in the starter set (see clause 5), are to be used in preference to presentation control functions, unless this is deemed particularly inconvenient.

4.7 Termination of a character set specification

A character set specification may be terminated either by an end-of-file marker (as recognized by the receiving system) or by the following line of text where the first two characters are question marks:

??END OF SPECIFICATION

NOTE. Care should be taken to ensure that the end of specification line does not occur anywhere before the end of the character set specification.

Table 1. Basic set of text characters	
Symbol	Name or description
	Space
A to Z	Capital letters
a to z	Small letters
0 to 9	Decimal (arabic) digits
&	Ampersand
'	Apostrophe
(	Left parenthesis
)	Right parenthesis
*	Asterisk
+	Plus sign
,	Comma
-	Hyphen
.	Full stop, period
/	Solidus
:	Colon
;	Semicolon
<	Less than sign
=	Equals sign
>	Greater than sign
?	Question mark
!	Exclamation mark
%	Per cent sign
[	Left square bracket
]	Right square bracket

5 Starter set of markup tags

5.1 Introduction

5.1.1 The following describes a starter set of typographic markup tags designed to provide a useful set for those who do not wish to design their own set (or use some pre-defined set outside this British Standard).

Those wishing to use a different set of markup tags should specify their markup tags according to the method of clause 6.

The markup tags of the starter set are grouped according to their meaning into the following groups:

- (a) markup tags whose function is to specify the style of characters (see 5.4.2);
- (b) markup tags whose function is to specify the size of characters (see 5.4.3);
- (c) markup tags whose function is to specify the position of one character relative to another (see 5.4.4);
- (d) markup tags whose function is to specify characters to be either superior or inferior to preceding text (see 5.4.5);
- (e) markup tags whose function is to specify the arrangement of characters into lines of text (see 5.4.6);
- (f) markup tags whose function is to specify the length of lines of text (see 5.4.7);
- (g) markup tags whose function is to specify the position of one line of text relative to another (see 5.4.8);
- (h) markup tags whose function is the creation of special graphic effects (see 5.4.9):
 - (1) rules;
 - (2) leaders.

5.1.2 There are three kinds of markup tag: typographic markup, generalized (structural, SGML) markup and symbol reference. The starter set contains only typographic markup tags.

NOTE. This guide does not describe the use of generalized markup tags or symbol reference tags, nor does it specify any within this starter set. However, it is recognized that many users of this guide will need to make simultaneous use of typographic and generalized markup and symbol reference tags. For this reason examples of generalized markup and symbol reference tags are included in appendices C and D.

5.2 Units of measurement

5.2.1 Identification of units of measurement

Markup tags in the starter set use, wherever appropriate, a variety of units of measurement. The units of measurement that are valid for each markup tag are given in 5.4.10. A specific choice of units of measurement is indicated in a markup tag by the use of one- or two-character abbreviations. These acronyms are given in 5.2.2. Their use in the markup tags of the starter set is described in 5.4.

5.2.2 Classification of units of measurement

Units of measurement are classified into the following groups:

- (a) large absolute units:

cm = centimetres
in = inches
pi = picas
ci = ciceros

NOTE. Although not expressly forbidden, fractional values should not be used with values expressed in picas or ciceros, as the meaning could be misinterpreted. For example, 12.9 could be misinterpreted as 'twelve picas and nine points', rather than 'twelve and nine-tenths picas'.

- (b) small absolute units:

mm = millimetres
pt = Anglo-American points
di = Didot points

- (c) horizontal relative units:

ru = system-dependent relative units, being proportional to the current nominal type width
em = ems, being the horizontal space equal to the value of the current nominal type width

- (d) vertical relative units:

ln = lines, being the vertical space equal to the value of the current interlinear space.

5.2.3 Relationship between units of measurement

The numerical relationships between the units of measurement are as follows:

- (a) 1 pi = 12 pt;
- (b) 1 ci = 12 di;
- (c) 1 cm = 10 mm.

No other exact relationships are recognized.

NOTE 1. The relationship between an em and a relative unit is system-dependent.

NOTE 2. The relationship between a pica and an inch is subject to slight variation between typesetting systems, contrary to the common assumption that there is a strict relationship of 6 pi (72 pt) to the inch.

5.2.4 Sub-units

The small absolute units (mm, pt, di) are also called sub-units when used in combination with the related large absolute units (cm, pi, ci) to express fractions of the latter.

Sub-units may be used either alone or in combination with the large absolute units to which they are related (see 5.2.3). The following combinations of units and sub-units are valid:

- (a) picas and Anglo-American points, e.g. 12 pi 6 pt;
- (b) ciceros and Didot points, e.g. 6 ci 2 di;
- (c) centimetres and millimetres, e.g. 10 cm 5 mm.

All other combinations of units and sub-units are invalid.

5.2.5 Decimal values

All numerical units, unless otherwise specified, may take values specified using decimal numbers. This guide makes no restriction on the number of digits that are valid before or after a decimal point (see 6.4). However, individual typesetting systems will have their own restrictions on the use of decimal values and the range of values allowed for each typesetting function.

5.3 Delimiters

5.3.1 General

Although only typographic markup tags are defined within this guide, the other kinds of tag (generalized markup and symbol reference) should be delimited in accordance with this guide if the resulting marked-up text is to conform to this guide.

Except when otherwise described in 6.3, the delimiters for tags should be as defined below:

Kind of tag	Opening delimiter	Closing delimiter
Typographic markup	[	]
Generalized markup	<	>
Symbol reference	&	;

NOTE. The delimiters for generalized markup tags and symbol reference tags are those specified in BS 6868.

5.3.2 Examples of delimiters in use

The following is a typographic markup tag in the starter set meaning ‘select roman fount of the current type family’:

[ro]

The following is a generalized markup tag that, in a given application, may be chosen to mean ‘the start of a new paragraph’:

< p >

The following is a symbol reference tag that, in a given application, may be chosen to mean ‘paragraph mark’:

¶

5.3.3 The validity of delimiters and markup tags

The characters [] < > &; or others specified as delimiters in their place (see 6.3) will only be recognized as delimiter characters where they form part of a valid markup tag. Any invalid markup

tag, i.e. one that has not previously been specified and does not occur in the starter set (see 6.7), will be treated as text and will not be recognized as an error.

NOTE. If invalid markup tags are treated as errors, these should be reported by the typographer in such a way as to give precise information on the nature of the error.

5.3.4 Omission of delimiters

All open and close delimiters should be keyed in markup tags except in such cases where markup tags are concatenated (run together) by the removal of intervening close and open delimiters. Concatenation is only valid for a very limited number of combinations of markup tags, all of which are specified under the definition of the relevant markup tags in 5.4.

NOTE 1. For general information on the use of concatenation see 6.8.

NOTE 2. Markup tag schemes that allow the omission of end delimiters, e.g. *i for italic, are not compatible with this guide.

5.3.5 Case sensitivity

When using markup tags of the starter set, letters of the alphabet may be capital or lower-case letters, or a mixture of the two.

If case sensitivity is specified using the methods of clause 6 (see 6.5), only markup tags containing lower-case letters are valid.

5.4 Typographic markup tags

5.4.1 General information

5.4.1.1 Markup tags required before any text

The following markup tags should be entered before any text:

- (a) a markup tag defining the primary type family (see 5.4.2.2);
- (b) a markup tag selecting a fount (see 5.4.2.7 to 5.4.2.12);
- (c) a markup tag defining the nominal type size (see 5.4.3.2);
- (d) a markup tag defining the measure (see 5.4.7.2);
- (e) a markup tag defining the interlinear space (see 5.4.8.2).

NOTE. If any of the above were excluded, the typesetting system would not have sufficient information to begin typesetting the text.

5.4.1.2 Functional scope of tags

Unlike generalized markup tags, which either implicitly or explicitly function in pairs, typographic markup tags mostly function individually.

Unless otherwise indicated, the scope of function of a typographic markup tag is all text that follows it until the occurrence of another markup tag whose function is to alter the same characteristics that were modified by the first tag.