

STANDARD ST.32

Version 1.0

RECOMMENDATION FOR THE MARKUP OF PATENT DOCUMENTS USING SGML (STANDARD GENERALIZED MARKUP LANGUAGE)

Revision adopted by the PCIPI Executive Coordination Committee at its seventeenth session on November 24, 1995

TABLE OF CONTENTS

- Introduction
- Definitions
- Character sets
- References
- Requirements of the recommendation
 - Part 1: SGML markup for common text
 - General text
 - **SGML tags: description and usage**
 - **Document structure**
 - **Miscellaneous**
 - **Page structure tags**
 - **Lists**
 - **SGML tags: description and usage**

Introduction

1. This Recommendation is an application of International Standard ISO 8879:1986, *Information Processing - Text and Office Systems - Standard Generalized Markup Language (SGML)*; which is under review.
2. This Recommendation provides for the exchange of patent documents in machine-readable form on any exchange medium in a hardware-, software- and layout-independent format. Such independence of the representation of the contents of a document from their intended uses is achieved by using International Standard ISO 8879:1986, Information processing - Text and office systems - Standard Generalized Markup Language (SGML), to define generic identifiers which are in turn used to mark the logical structure of each patent document.
3. International Standard ISO 8879:1986 cannot be used *per se* as the basis for document processing. That is not the intention of the standard. Instead, ISO 8879 "standardizes the application of generic coding and generalized markup concepts. It provides a coherent and unambiguous *syntax* for describing whatever a user chooses to identify within a document" (ISO 8879:1986 page 2). The choice of tags, that is, the *semantics* to which the syntax applies, is left to the user.
4. Therefore, this Recommendation defines generic identifiers or "tags" for marking the logical elements of a patent document. The logical elements of a patent document are of two types: common text and patent-specific content.
5. Under the terms of International Standard ISO 8879:1986 any tags may be used in a particular document so long as the semantics are defined in an accompanying document type definition (DTD). It is conceivable that a patent issuing authority may choose different tags than those specified in this Recommendation. So long as the tags were defined in the accompanying DTD, the document could be presented to a user on a system designed to read SGML documents. However, documents which use a DTD that differs from that specified below cannot be considered to be in compliance with this Recommendation even if they are in compliance with ISO 8879:1986.
6. Markup in compliance with this Recommendation is independent of layout and formatting. Decisions regarding layout and formatting must be made at the time a document is presented for reading, either on a display screen or on paper. It is at the time of presentation that, for example, text which has been marked as emphasized (bold, italic, etc.) is rendered in an available font which has more or less the desired appearance. It is at the time of presentation that the size of the display page (screen or paper) is determined. Many such decisions which map the generic identifiers in a document to the capabilities of a particular physical display device (whether screen or paper) determine, for example, how many characters will fit on one line or how much text will fit on a display page. As a result, the document may not have exactly the same physical appearance when it is presented on different display devices. This Recommendation does not address issues concerned with mapping generic identifiers to a particular display device. It can be expected that in the future two standards may be applied in this area: Standard Page Description Language (SPDL) ISO/IEC DIS 10180 and Document Style Semantics and Specification Language

(DSSSL) ISO/IEC DIS 10179.

7. Markup in compliance with this Recommendation should facilitate importing large sets of documents into a database. In fact, the extensive list of tags for patent bibliographic data will make it possible for database vendors to more easily distinguish various information elements with higher precision than has been possible in the past. This Recommendation does not address issues concerned with mapping generic identifiers to database fields.
8. This revision of ST.32 shall be referenced as version 3 (1995). This is to distinguish it from previous versions, which may still be used for data exchange but, if so, **must** be referenced as: version 1 (October 1987) or version 2 (September 1990). The relevant DTD may then be applied to a specific version for processing, parsing, etc. In addition it is possible to reference the DTD to be used as an attribute to any patent document, the default being the latest version of ST.32. It is, of course, recommended to update files to this latest version of ST.32 for data exchange.

Definitions

9. The expression **patent document** includes patents for invention, plant patents, design patents, utility certificates, utility models, documents of addition thereto and published applications therefor. (Refer also to WIPO [ST.16](#): *Recommended Standard Code for the Identification of Different Kinds of Patent Documents*)
10. **Common text** refers to logical elements that could occur in any type of industrial property information or in any kind of document, for example, paragraphs, footnotes, subscripts, special characters, lists, embedded images, tables, chemical formulae, mathematical formulae, etc. Tags for common text data are specified and described in Part 1 (the DTD is in Annex B).
11. **Patent-specific content** refers to logical elements that ordinarily occur only in patent documents, for example, inventor's name, patent number, issuing authority, priority data, classification symbols, etc. In short, any of the information elements identified in WIPO Standard [ST.9](#), *Recommendation Concerning Bibliographic Data on and Relating to Patents and SPCs*, as well as some others. Tags for patent bibliographic data are specified and described in Part 2 (the DTD is in Annex B).
12. **Markup** is defined as text that is added to the content of a document and that describes the structure and other attributes of the document in a non-system-specific manner, independently of any processing that may be performed on it. Markup includes document type definitions (DTDs), entity references, and descriptive markup (tags).
13. A **document type definition (DTD)** formally defines:
 - the names of all the logical elements that are allowed in documents of a particular type;
 - how often each logical element may appear;
 - the permissible contents for each logical element;
 - attributes (parameters) that may be used with each logical element;
 - the correct sequence of logical elements;
 - the names of all external and pre-defined entities that may be referenced in a document;
 - the hierarchical structure of a document;
 - the features used from the SGML standard.

A DTD defines the vocabulary of the markup for which SGML defines the syntax. The complete set of tags that may be found in a particular document are listed and formally defined in its DTD which must accompany the document. Each document in a large set of documents which share the same DTD, that is, documents which are of the same type, usually incorporates the DTD by reference.
14. An **entity** is content that is not part of the text stream in a document but which is incorporated into the text stream by reference to its name. In patent documents, for example, images are external entities. Entity references can also be used to code instances of characters not found in the 'declared' character set (see Character Sets below).
15. **Tags** define a document's logical structure by labelling elements of the document's content using the generic identifiers declared in the DTD.
16. The **hierarchy** of SGML tags used in this Recommendation follows the general structure of a patent document. The level in the hierarchy is indicated by the appropriate SGML tag describing a generic logical element. A generic logical element is a component of the text such as the entire document, a specific sub-document, a paragraph, a list, etc. Each generic logical element is described by a start tag and end tag.

level

sgml tag (example)

Document	<PATDOC>
. Sub-document	<SDOXX>
. . Text Component (Paragraph)	<P>
. . . Text Element (Subscript)	<SB>
. . . . Character	
. . . End	</SB>
. . End	</P>
. End	</SDOXX>
End	</PATDOC>

17. International Standard ISO 8879:1986 defines an **abstract syntax** and a **reference concrete syntax**. The reference concrete syntax for SGML tags is as follows:

Start	End
Tag	Tag

This is text that will appear emphasized as bold ...

Where

- < is the opening delimiter for Start Tags (1 character)
- </ is the opening delimiter for End Tags (2 characters)
- > is the closing delimiter for both Start Tags and End Tags (1 character)
- B is the generic identifier of this particular tag, defined in the DTD

A generic identifier is a name that identifies a generic logical element. The text between the start tag and the end tag is a specific instance of the generic logical element. Depending upon the generic identifier, parameters may be required. In the description of the various tags in this Recommendation, parameters are referred to as "attributes" in conformance with ISO practice. For an explanation of the relationship between reference concrete syntax and abstract syntax, see International Standard ISO 8879:1986.

18. The following is a brief example of SGML markup:

```
<PATDOC>
<SDOBI LA=EN>
<B100> (Minimum bibliographic tags would need to be added here)
</SDOBI>
<SDODE LA=EN>
<H LVL=0>FABRIC SOFTENER COMPOSITION</H>
<H LVL=1>TECHNICAL FIELD</H>
...
<H LVL=1>SUMMARY OF THE INVENTION</H>
<P N=11>The present invention relates to an aqueous fabric softener composition comprising:
<SL>
<LI>(A) from 1% to 50% by weight of ... cyclic amines of the formula
<EMI FILE="92102108" ID="2.1" HE=30 WI=55 TI=CF>
```

wherein n is 2 or 3, R<SB>1</SB> and

...

<L>(B) from 3% to 20% by weight of (A)

...

</SL>

</SDODE>

</PATDOC>

In the example above <EMI FILE="92102108" ID="2.1" HE=30 WI=55 TI=CF> refers to a chemical structure which has been scanned as an image and which will be imbedded in the text at this point at the time of presentation. <PATDOC> and </PATDOC> mark the beginning and end of a patent respectively. The other tags in the example are explained below and there are more extensive examples in Annex D.

Character sets

19. The data content of the majority of documents, including patents, consists of data characters. The data characters could be in any language consisting of many types of character ('character' is used in its broadest sense here to include graphical symbols). In this recommendation only one coded character set is referenced: ISO 646. This is probably the most common **system independent** character set in use today. Characters not in this code set should be represented by public entity references - preferably those contained in ISO 8879 - these are referenced in the DTD in Annex B. Note that other character sets and character entity references are possible. It is not recommended to use the code pages contained in WIPO [ST.31](#) since these can lead to problems in data interchange, are not easily maintained and are not as commonly used and accepted as the ISO 646 code page.

References

20. The following documents are of fundamental importance to this Recommendation:

International Standard ISO 8879:1986, *Information processing - Text and office systems - Standard Generalized Markup Language (SGML)*;

Technical Report ISO/IEC/TR 9573:1988(E) *Information processing - SGML support facilities - Techniques for using SGML*;

International Standard ISO 639:1988, *Code for the Representation of Names of Languages*.

International Standard ISO 646:1991, *Information Processing - ISO 7-bit coded character set for information interchange*.

WIPO Standard [ST.3](#), *Recommended Standard Two-Letter Code for the Representation of States, Other Entities and Intergovernmental Organizations*;

WIPO Standard [ST.9](#), *Recommendation Concerning Bibliographic Data on and Relating to Patents and SPCs*;

WIPO Standard [ST.16](#), *Recommended Standard Code for the Identification of Different Kinds of Patent Documents*.

21. For additional information concerning SGML the following publications may be of interest (please note there is now a considerable amount of literature, books and periodicals, on SGML, as well as many user groups, the list below is only a small selection):

American National Standards Institute. *Electronic manuscript preparation and markup*. (Z39.59). Transaction Publishers : New Brunswick (USA) and London, 1991. ISBN 0887389457.

Association of American Publishers. *Electronic Manuscript Series : Author's guide to electronic manuscript preparation and markup; Reference manual on electronic manuscript preparation and markup; Markup of mathematical formulas; Markup of tabular material*. Dublin, Ohio : Electronic Publishing Special Interest Group (EPSIG), 1989.

Bryan, Martin. *SGML : an author's guide to the Standard Generalized Markup Language (SGML)*. Wokingham : Addison-Wesley, 1988. ISBN 0201175355.

Goldfarb, Charles F. *The SGML handbook*. Oxford : Oxford University Press, 1990. ISBN 0 19 853737 9.

Van Herwijnen, Eric. *Practical SGML. 2nd.ed.* Dordrecht : Kluwer Academic Publishers, 1994. ISBN 0792394348

Requirements of the recommendation

22. Documents which conform to this Recommendation shall be marked up in conformance with:
- International Standard ISO 8879:1986, Information Processing - Text and Office Systems - Standard Generalized Markup Language (SGML);
 - the DTD contained in Annex B.
23. Documents which conform to this Recommendation shall use the *reference concrete syntax* defined in International Standard ISO 8879:1986. See also Annex A: *SGML Declaration for Patent Documents*.
24. The DTD contained in Annex B shall be provided separately from the individual documents in the collection of documents to which it applies.
25. Each document to which the DTD in Annex B applies will incorporate the DTD by reference.
26. Reference to the DTD contained in Annex B shall be made by use of its "public name" which has been [will be] registered with the appropriate international authority and is declared below in Annex B.
27. No document in conformance with this Recommendation shall refer to or incorporate by reference a DTD 1) for which a public name has not been registered with the appropriate international authority; 2) which does not appear in this Recommendation.
28. It may happen that some particularly unusual document contains some text or image portion(s) which cannot be rendered for the end user with adequate fidelity, in the judgement of the issuing authority, without the introduction of one or more logical elements not contained in Annex B. In that event:
- 28.1. The issuing authority shall provide constructive notice to end users that some documents contain exceptional elements. Where possible, the exact identification of such documents shall be provided, either as a list of document numbers or contiguous ranges of document numbers.
- 28.2. The issuing authority shall make every attempt to have the required logical element(s) introduced into the appropriate DTD contained in the appropriate section of this Recommendation, so that other issuing authorities may take advantage of them, and so that presentation system vendors may take account of them in preparing presentation software and hardware.
- 28.3. The issuing authority may, at its discretion, include the required logical element(s) in a supplementary DTD which is incorporated by reference into the DTD(s) that apply to the document(s) in question until such time as the elements are incorporated into this Recommendation.
- 28.3.1. A supplementary DTD shall not be incorporated directly into the document(s) to which it applies.
- 28.3.2. A supplementary DTD shall not contain any duplicate logical elements included in the DTD contained in ST.32, Annex B.
- 28.3.3. If a supplementary DTD is provided, constructive notice shall be given to the end user to that effect.

Part 1: SGML markup for common text

The tags described in this part of ST.32 indicate text portions that are not specific to any one type of industrial property information and may therefore be used in any document conforming to ST.32.

General text

TABLE OF SGML TAGS

TAG	NAME	DESCRIPTION
	Bold	Indicates the beginning of text to be highlighted at the time of presentation by using a bold typestyle. An end tag is required.
<BAI>	BAlkaku	Indicates Japanese text portion to be highlighted using an expanded font. An end tag is required.
<BCHG>	Beginning of a CHanGe	Indicates the beginning of a change in bibliographic data only.

		Attributes required. It is an empty element which should be followed by <ECHG>.
 	line BReak	Indicates the position in the text at which a line break occurs. No end tag is necessary.
<CHF>	CHaracter Fraction	Indicates a character construct consisting of two or more characters in a 'fraction type' construct. Use with the <CHFBR> tag. An end tag is required.
<CHFBR>	CHaracter Fraction BReak	Indicates the break point in a character 'fraction' construct consisting of two or more characters in a 'fraction type' construct. No end tag is necessary.
<CHG>	CHanGe	Indicates the beginning of a change (not in bibliographic data). Attributes required. An end tag is required
<DP>	Document Page	Indicates the beginning of a new page. The attribute N= is required. No end tag is necessary.
<ECHG>	End of a CHanGe	Indicates the end of a change in bibliographic data only. Attributes required. It is an empty element which should be preceded by <BCHG>.
<FLA>	FLoating Accents	This indicates a character enhanced with a particular attributing feature. An end tag is required.
<FLAC>	FLoating ACcents	This indicates the attributing feature in a floating accent construct. No end tag is necessary.
<FOO>	FOOtnote	Indicates a footnote. Attributes required. An end tag is required.
<FOR>	FOotnote Reference	Indicates a reference to a previous footnote. Attributes required. An end tag is required.
<H>	Heading level	Indicates a separate text portion that precedes text parts, for example, paragraphs. An end tag is required.
<HAN>	HANKaku	Indicates Japanese text portion to be highlighted using a compressed font. An end tag is required.
<I>	Italic	Indicates the beginning of text to be highlighted at the time of presentation by using an italic typestyle. An end tag is required.
<LTL>	LiTeraL	Indicates the beginning of text in which the space, indents, line endings, etc., should be preserved as keyed in the original document. An end tag is required.
<O>	'Over' embellishments	Indicates the beginning of text to be covered by an over, or mid, embellishment of a particular designated style (attribute) at the time of presentation. An end tag is required.
<P>	Paragraph	Indicates a text portion known as a paragraph and implies that the text will begin on a new line. No end tag is necessary.
<PATDOC>	PATent DOCument	Indicates the beginning of a patent document instance (file). An end tag is required.
<PC>	Paragraph Continuation	Indicates a continuation of an interrupted paragraph. No end tag is necessary.

<PCL>	Page CoLumn	Indicates the beginning of a new column. The attribute N= is required. No end tag is necessary.
<PLN>	Page LiNe	Indicates the beginning of a new line. The attribute N= is required. No end tag is necessary.
<SB>	SuBscript	Indicates the beginning of text which is to be placed as a subscript to the preceding text outside mathematical formulae. An end tag is required.
<SDOxx>	Sub-DOcument	Indicates the beginning of a sub-document whose identity (xx) is included in the tag. An end tag is recommended.
<SP>	SuPerscript	Indicates the beginning of text which is to be placed as a superscript to the preceding text outside mathematical formulae. An end tag is required.
<TXF>	TeXt Frame	This indicates a rectangular area of text of a page. No end tag is necessary.
<U>	Under embellishment	Indicates the beginning of text to be highlighted with an under embellishment of a particular style (attribute) at the time of presentation. An end tag is required.

TABLE OF ATTRIBUTES

ATTRIBUTE	NAME	DESCRIPTION
ALIGN	ALIGN	Alignment of data.
CY	CountrY code	Indicates country code based on ST.3
DATE	DATE	Date : in the format YYYYMMDD
DNUM	Document Number	Identifier of document (publication or application number)
DTD	Document Type Definition	Version number of the DTD.
FILE	File name	The file name of a patent document or image file.
FN	Footnote Number	Unique identification for a footnote
FNREF	Foot Note REference	Unique identification for a footnote reference
FONT	FONT	Font used in text frames
FR	FRame	Frame number within a page
HE	HEight	Height of images in mm
ID	IDentifier	Has various parameters depending on tag
KIND	KIND	Kind of document based on ST.16
LA	LAnguage	Indicates sub-document language
LS	Line Spacing	Line spacing in text frames

LVL	LeVeL	Indicates the level of a heading
LX	X coordinate	X coordinate of image in 1/10mm
LY	Y coordinate	Y coordinate of image in 1/10mm
N	Number	Indicates numbers for paragraphs, pages, etc.
POS	POSition	Indicates various parameters depending on the tag
SIZE	SIZE	Font size in text frames
STATUS	STATUS	Indicates status of patent document and/or change
STYLE	STYLE	Indicates the style of various attributes, for example, over characters, etc.
TYPE	TYPE	Type of embellishment
WI	Wldth	Width of images in mm

SGML tags: description and usage

Document structure

1. <PATDOC> : PATent DOCument

This is the mandatory identifier with which every patent document must start. An end tag is required.

Required Attribute(s):

None.

Optional Attribute(s):

FILE=name Where 'name' is the name of the patent document file, which contains the document instance.

STATUS= Status of the patent document, eg. contains changes, republished, deleted, withdrawn, etc.

Note: It is recommended that the following optional attributes should be used only when the mandatory tags, giving document identification, contained in the <SDOBI> sub-document, are not used. This may be the case, for example, when only partial information is exchanged between offices.

CY=xx Where xx is the country or organisation, according to WIPO [ST.3](#), publishing or issuing the patent document. <B190>

DATE=YYYYMMDD Date of publication. <B140>

DNUM=n Where n is the document number, usually the publication number but may also be the application number. <B110> or <B210>

KIND=xx Where xx is the kind of patent document code taken from WIPO [ST.16](#). <B130>

DTD=*n* Where *n* is the version number of the DTD applied to a particular patent document. The default is ST.32 Version 3 (1995).

DTD Syntax:

```
<!ELEMENT patdoc - -      (sdoabi,(sdoab*&sdode?&sdocl*&sdodr?&sdosr?))
                          +(%floats;)
>
<!ATTLIST patdoc cy       CDATA    #IMPLIED    -- Country, organis. St.3    --
                  dnum    CDATA    #IMPLIED    -- Identification number    --
                  date    NUMBER  #IMPLIED    -- date of publication      --
                  file     CDATA    #IMPLIED    -- file identification      --
                  kind     CDATA    #IMPLIED    -- Kind of patent St.16     --
                  status   CDATA    #IMPLIED    -- Status of the patent doc. --
                  dtd      NUTOKEN #IMPLIED    -- Version NUMBER of DTD    -->
```

Examples:

```
<PATDOC><SDOBI>Here is a WIPO Patent Document (other tags would normally be included)</SDOBI></PATDOC>
```

```
<PATDOC FILE92101123 CY=EP DATE=19921212 DNUM=0500111 KIND=A1>
```

```
<SDOBI>Here is a European Patent Office application with a search report (A1)(other tags would normally be
included)</SDOBI></PATDOC>
```

2. <SDOxx> : Sub-Document tags

This is the mandatory identifier with which every sub-document must start. An end tag, although optional, is recommended.

Where *xx* = sub-document identifier

Possible sub-documents are:

<SDOAB> Abstract

<SDOBI> Bibliographic data

<SDOCL> CLaims

<SDODE> DEscription

<SDODR> DRawings

<SDOSR> Search Report

Required Attribute(s):

None.

Optional Attribute(s):

CY=country code Indicates the country where the sub-document "CLAIMS" especially relate to, abbreviated in accordance with WIPO Standard ST.3 country code.

LA=language code Indicates language of the sub-document in accordance with International Standard ISO 639:1988.

STATUS= Status of the patent sub-document, eg. contains changes, republished, deleted, withdrawn, etc.

DTD Syntax:

```
<!ELEMENT sdo bi - o (B000?,B100,B200?,B300?,B400?,B500?,B600?,B700?,B800?,
                    B900?) +(bchg|echg) -- Bibliographic data -->
<!ELEMENT sdo ab - o ((h|p|pc|%img;)+) -- Abstract -->
<!ELEMENT sdo dr - o (emi+) -- Drawings -->
<!ELEMENT sdo de - o (h|p|pc|%img;)+ -- Description -->
<!ELEMENT sdo cl - o (h|p|%lst;)+ -- Claims -->
<!ELEMENT sdo sr - o ((B510?,B520?,B560?,B580?)|(emi+)) -- Search report -->
<!ATTLIST (sdo bi|sdo ab|sdo de|sdo cl|sdo dr|sdo sr)
          la NAME #IMPLIED -- language(ISO 639) --
          cy NAME #IMPLIED -- country code --
          status CDATA #IMPLIED -- Status of the sub-doc. -->
```

Examples:

```
<SDOBI>
<B500><B542>Here is bibliographic data - it should contain other tags also</B500>
</SDOBI>
<SDOCL LA=F>
<OL>
<LI>Dispositif de reséquence ment (RU) pour un noeud d'un système de commutation de cellules, chaque cellule étant constituée
d'un nombre variable de sous-cellules ayant une longueur fixe, ce noeud comportant .....
<LI> ...
</OL>
</SDOCL>

<SDOAB LA=D>
<P>Die vorliegende Erfindung betrifft Impfstoffe auf Basis von Bovinen Herpesviren des Typs 1 (BHV-1) die Änderungen in
Bereichen ihres Genoms enthalten, die für nicht-essentielle Teile essentieller Proteine kodieren. Mit Hilfe dieser Impfstoffe können
geimpfte von nicht-geimpften Rindern unterschieden werden. Die Erfindung betrifft ferner Verfahren zur Isolierung und Herstellung
der geänderten BHV-1 Stämme, Isolierung und Herstellung der geänderten Proteine und Peptide.
</SDOAB>

<SDOCL LA=D CY=AT>
<OL>
```

Mikroorganismus DSM 7329 und DSM 7330.

Verfahren zur Herstellung von L- α -Aminosäuren durch enzymatische Umsetzung eines D-, L- und/oder D,L-5-monosubstituierten Hydantoins und/oder einer D-, L- und/oder D,L-N-Carbamoyl- α -aminosäure,

</SDOCL>

3. <CHG> : CHanGe

This indicates data which has been 'changed' (it could also indicate the original text). An end tag is required.

Required Attribute(s):

DATE=YYYYMMDD Indicates the date on which the text was changed.

STATUS= Indicates the status of the change, the value of this attribute has been left open but one letter codes are recommended, eg. A = amended text, D = deleted text, O = Original text.

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT chg    - - (h|p|pc|(%ptext;))*           -- Change text           -->
<!ATTLIST chg   date      NUMBER          #REQUIRED    -- Date of change text     --
                status    CDATA           #REQUIRED    -- Status of the change    -->
```

Example:

<P><CHG DATE=19950321 STATUS=A>This data was amended on 21 March 1995</CHG>

4. <BCHG> : Beginning of a CHanGe

This indicates bibliographic data which has been 'changed' (it could also indicate the original text). It is an empty element - it should be followed by <ECHG>.

Required Attribute(s):

DATE=YYYYMMDD Indicates the date on which the text was changed.

STATUS= Indicates the status of the change, the value of this attribute has been left open but one letter codes are recommended, eg. A = amended text, D = deleted text, O = Original text.

Optional Attribute(s):

None

DTD Syntax:

```
<!ATTLIST bchg date NUMBER #REQUIRED -- Date data changed --
                status CDATA #REQUIRED -- Status of the change -->
```

Example:

```
<B235><BCHG DATE=19960321 STATUS=A><DATE>19960321</DATE><ECHG></B235>
```

5. <ECHG> : **End of CHanGe**

This indicates the end of data which has been 'changed' in bibliographic data (it could also indicate the original text). It is an empty element - it should be preceded by <BCHG> .

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT echg - o EMPTY -- End of changed bibliographic data -->
```

Example:

```
<B235><BCHG DATE=19960321 STATUS=A><DATE>19960321</DATE><ECHG></B235>
```

6. <H> : **Headings**

This indicates levels of headings which may be treated differently. An end tag is required.

Required Attribute(s):

None.

Optional Attribute(s):

LVL=n Indicates the level of the heading

ALIGN Indicates the alignment of the header which may be centre, left, right - left is the default.

DTD Syntax:

```

<!ELEMENT h      - - (%ptext;)+          -- Header      -->
<!ATTLIST h      lvl NUMBER #IMPLIED    -- Header level --
                align (%align;) "left"  -- alignment   -->

```

Examples:

```
<H>This is a default heading</H>
```

```
<H LVL=0>This is the title heading</H>
```

```
<H LVL=1>This is a sub-section heading</H>
```

7. <P> : Paragraphs

This indicates a text portion commonly known as a paragraph. No end tag is necessary.

Required Attribute(s):

None.

Optional Attribute(s):

N=nnnnnn Consisting of a 6-digit sequence number indicating every paragraph in a document or sub-document. Leading zeros may be dropped.

ALIGN= Indicates the alignment of the paragraph which may be centre, left, right - left is the default.

Example:

```
<P>First text paragraph . <P>Second text paragraph.
```

```
<P N=1>First text paragraph . <P N=2>Second text paragraph.
```

DTD Syntax:

```

<!ELEMENT p      - o (%ptext;)+          -- Paragraph elements -->
<!ATTLIST p      n    NUMBER #IMPLIED    -- Reference number   --
                align (%align;) "left"  -- alignment          -->

```

8. <PC> : Paragraph Continuation

This indicates an interruption in a paragraph, for example, by a figure, table, etc. The existing paragraph should be continued. No end tag is necessary.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT pc      - o (%ptext;)+          -- Paragraph continuation      -->
```

Example:

```
<P N=12>Here starts a new text paragraph, it contains an EMI:
```

```
<EMI ID='2.1' HE=10 WI=20 TI=CF>
```

```
<PC>and continues without paragraph formatting ...
```

9. **
 : BReak**

This indicates a line break in general text. No end tag is necessary. Whether and how the break tag is interpreted at the time of presentation is not specified in this Recommendation. Note that this tag should not be used in mathematical formulae where **<BREAK>** is used.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT br      - o EMPTY              -- Line break                      -->
```

Example:

This line must break here

and also break here

but that's all for this paragraph.

```
<P>This line must break here<BR>and also break  
here<BR>but that's all for this paragraph.
```

Note: the above example assumes that the break tag is interpreted at the time of presentation as forcing a line break in the text. Other interpretations are possible.

10. **<FOO> : FOOtnotes**

This tag identifies a text portion which is the contents of a footnote. The footnote should be inserted in the text stream at the point where it is first referred to. The presentation software will cause the footnote to appear, usually, at the bottom of the page. An end tag is required.

Required Attribute(s):

FN=nnnn.nn Consisting of a 4-digit sequence number indicating the page number of the original document on which the footnote occurred and a 2-digit sequence number indicating the sequence of footnotes on that particular page. Optionally, it may be replaced by a sequential numbering of footnotes within a document, in which case use FN=nnnnnn. Either form is valid. It must be a unique reference in the document. Leading zeros may be dropped.

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT foo - - (%ptext;)+ -- Footnotes -->
<!ATTLIST foo fn NUTOKEN #REQUIRED -- Footnote id. -->
```

Example:

... text ***<FOO FN='10.1'>*** This is the text of the footnote - to be placed at the foot of a page - note that the asterisk "*" is also part of this footnote**</FOO>**

Note: The indicator, in this case "*", is NOT inserted by application software, as is normal, because in patent documentation it is often not possible to change data submitted by a patent applicant.

11. **<FOR>** : **F**ootnote **R**eference

This indicates from which point(s) in a document a footnote is referenced. An end tag is required.

Required Attribute(s):

FNREF=nnnn.nn Consisting of a 4-digit sequence number indicating the original page number on which the footnote occurred and a 2-digit sequence number indicating the sequence of footnotes on that particular page. This attribute should contain exactly the same value as the attribute of the referenced footnote (FN=) . Optionally, it may be replaced by a sequential numbering within a document, in which case use FNREF=nnnnnn. Either form is valid. Leading zeros may be dropped.

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT for    - - (%ptext;)+          -- Footnote reference      -->
<!ATTLIST for   fnref  NUTOKEN  #REQUIRED -- Footref id.              -->
```

Example:

```
text<FOR FNREF='10.1'*></FOR> ...
```

Note: At the time of presentation this should result in the SAME footnote as first appeared on page 10 of the original document being produced on the page where <FOR> is used. This may occur, for example, if there is a page break during processing between the two footnote references which were originally on the same page.

TEXT 'HIGHLIGHTING' MARKUP

Note: The following codes: , <BAI>, <HAN>, <I>, <O>, <U>, <SB> and <SP> may be regarded as tags which can be used to mark characters, words, phrases, etc. as 'highlighted', that is emphasised in some way. In other instances they may be replaced by a 'pure' SGML tag such as <HPn>, highlighted phrase, where n is the numeric value assigned to a particular form of highlighting which is determined at the time of presentation (bold, italic, etc.). However, for patent documents, for the purposes of readability, it is recommended that the codes below be used instead. (Highlighted phrase identifiers are not contained in the DTD).

12. : **Bold**

This indicates a text portion to be highlighted as bold. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT b      - - (%ptext;)+ -(b)          -- Bold typeface          -->
```

Example:

This text is bold This text is bold

13. <BAI> : **BAlkaku**

This indicates a Japanese text portion to be highlighted using an expanded font. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT bai - - (%ptext;)+ -(bai|han) -- Expanded font -->
```

14. <HAN> : HANKaku

This indicates a Japanese text portion to be highlighted using a compressed font. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT han - - (%ptext;)+ -(han|bai) -- Compressed font -->
```

15. <I> : Italic

This indicates a text portion to be highlighted as italic. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT i - - (%ptext;)+ -(i) -- Italics -->
```

Example:

This text is italic

<I>This text is italic</I>

16. <o> : 'Over' embellishments

The over-character tag is used to identify parts of text over which special accents or diacritical marks are to be placed.

Note: the 'mark' could also be placed mid character. In mathematical formulae use <OV> . An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

POS= The position attribute defines the position of the mark and takes one of the values: above or mid. The default value is above.

STYLE= The style attribute defines the style of the mark. It takes one of the values: single, double, triple, dash, dots, or bold. The default value is single.

TYPE= The type attribute defines the type of the mark to be used. It takes one of the values: dot, dotdot, dot3, dot4, tie, tiebrace, hat, hacek, acute, grave, cedil, ring, macron, ogonek, dblac, breve, tilde, vec, rvec, dyad, circle, caret, prime, dprime, plus, none, bar. The default value is bar.

Note: Not all combinations of type and style attribute values would normally be used, eg typetilde, style=dots.

DTD Syntax:

```
<!ELEMENT o - - (%ptext;)+ -(o|ov) -- 'Over' embellishment -->
<!ATTLIST o pos (above|mid) "above" -- position of 'over' embellishment --
type (%type;) "bar" -- types of embellishment(default bar) --
style (%style;) "single" -- Line style (default single) -->
```

Examples:

<O>This text has a single continuous bar over the whole sentence</O>

The word example is covered by a dashed bar

The word <O STYLEDASH>example</O> is covered by a dashed bar.

17. <U> : **Under embellishments**

The under-character tag is used to identify parts of text under which special accents or diacritical marks may be placed - typically an underscore. In mathematical formulae use <OV POS=BELOW>. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

STYLE= The style attribute defines the style of the mark. It takes one of the values: single, double, triple, dash, dots, or bold. The default value is single.

TYPE= The type attribute defines the type of the mark to be used. It takes one of the values: dot, dotdot, dot3, dot4, tie, tiebrace, hat, hacek, acute, grave, cedil, ring, macron, ogonek, dblac, breve, tilde, vec, rvec, dyad, circle, caret, prime, dprime, plus, bar. The default value is bar.

Note: Not all combinations of type and style attribute values would normally be used, eg typetilde, styledots.

DTD Syntax:

```
<!ELEMENT u      - - (%ptext;)+ -(u|ov)      -- Underscore embellishment      -->
<!ATTLIST u      type (%type;) "bar"        -- types of embellishment(default bar) --
                style(%style;)"single"     -- Line style      (default single)   -->
```

Examples:

This text has a single continuous underscore for the whole sentence. <U>This text has a single continuous underscore (bar) for the whole sentence. </U>

The word example is underscored with a double bar. The word <U STYLE=DOUBLE>example</U> is underscored with a double bar.

18. <SB> : **SuBscript**

This indicates a text portion to be placed as a subscript (inferior) to the immediately preceding character. An end tag is required. See also the <SUB> tag used in mathematical formulae.

Required Attribute(s):

None

Optional Attribute(s):

POS= The position attribute takes one of the values: pre, mid or post, post being the default.

DTD Syntax:

```
<!ELEMENT sb      - - ((%hil;)|(#PCDATA))* -(fla) -- Subscript      -->
<!ATTLIST (sp|sb) pos (PRE|MID|POST) "POST"      -- Position (default post)  -->
```

Example:

H₂O H<SB>2</SB>O

19. <SP>: SuPerscript

This indicates a text portion to be placed as a superscript (superior) to the immediately preceding character. An end tag is required. See also the <SUP> tag used in mathematical formulae.

Required Attribute(s):

None

Optional Attribute(s):

POS= The position attribute takes one of the values: pre, mid or post, post being the default.

DTD Syntax:

```
<!ELEMENT sp      - - ((%hil;)|(#PCDATA))* -(fla) -- Superscript      -->
<!ATTLIST (sp|sb) pos (PRE|MID|POST) "POST"      -- Position (default post)  -->
```

Example:

Xⁿ⁻¹ X<SP>n-1</SP>

Miscellaneous

20. <CHF>: CHaracter Fraction constructs

This indicates 'fraction' constructs in general text. The alignment of 'numerator' and 'denominator' is centred by default. It should be used only in general text. An end tag is required.

See also the 'true' fraction tag, <FRAC>, used in mathematical formulae.

Required Attribute(s):

None

Optional Attribute(s):

ALIGN= Indicates the alignment of the numerator and/or denominator, which may be centre, left, right - centre is the default.

Example: (See below)

21. **<CHFBR> : CHaracter Fraction BReak**

This identifies the start of a character fraction 'denominator'. No end tag is necessary.

Required Attribute(s):

None

Optional Attribute(s):

STYLE= The style attribute defines the style of the mark preceding the character fraction denominator. It takes one of the values: single, double, triple, dash, dots, or bold. The default value is single.

TYPE= The type attribute defines the type of the mark to be used preceding the character fraction denominator. It takes one of the values: dot, dotdot, dot3, dot4, tie, tiebrace, hat, hacek, acute, grave, cedil, ring, macron, ogonek, dblac, breve, tilde, vec, rvec, dyad, caret, circle, plus, prime, dprime, bar, none. The default value is bar.

Note: Not all combinations of type and style attribute values would normally be used, eg typetilde, styledots.

DTD Syntax:

```
<!ELEMENT chf - - ((#PCDATA)|(%hil;))+, chfbr) -- character fraction -->
<!ATTLIST chf align (%align;) "centre" -- alignment --
<!ELEMENT chfbr - o ((#PCDATA)|(%hil;))+ -- character fraction break -->
<!ATTLIST chfbr type (%type;) "bar" -- types of embellishment(default bar) --
style (%style;) "single" -- Line style (default single) -->
```

Examples:

20 <CHF>20<CHFBR TYPE=NONE>D</CHF>

D

22. **<FLA> : FLloating Accents**

This indicates a character, or characters, enhanced with a particular attributing feature(s). It enables "composite" characters not in a character set to be composed from characters and character entity references. It should be used in combination with the <FLAC> tag. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT fla - - ((#PCDATA)|(%hil;))+, flac) -- Floating accent construct -->
```

23. <FLAC> : FLloating ACcent

This indicates the start of a floating accent to be placed above, mid, or below a base character, or characters, above is the default. It enables "composite" characters not in a character set to be composed from characters and character entity references. It should be used in combination with the <FLA> tag. No end tag is necessary.

Required Attribute(s):

None

Optional Attribute(s):

POS= The position attribute takes one of the values: above, mid or below, above being the default.

DTD Syntax:

```
<!ELEMENT flac - o ((#PCDATA)|(%hil;))+ -- Floating accent (upper part) -->
<!ATTLIST flac pos (ABOVE|MID|BELOW) "ABOVE" -- Position (default above) -->
```

Examples:

Å <FLA>A<FLAC>ˆ</FLA>

a <FLA>a<FLAC>•</FLA>

24. <LTL> : LiTeraL text

Indicates the beginning of text in which the space, indents, line endings, etc., should be preserved as keyed in the original layout. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

WI=nnn Width: 3-digit expression in millimetres.

DTD Syntax:

```
<!ELEMENT ltl - - CDATA -- Literal text -->
<!ATTLIST ltl wi NUMBER #IMPLIED -- Width in mm -->
```

Example:

	<LTL>
This text	This text
has a special	has a special
layout	layout
which must be	which must be
preserved	preserved
exactly	exactly
as entered.	as entered.
	</LTL>

Page structure tags

The following tags are specific to patent document processing and are to be used to indicate page structure in order to allow exact citation of pages, page numbers, columns and lines. For post processing of the data these tags can, of course, be ignored if required.

25. <TXF> : TeXt Frame

This indicates an area of text within a page of a document. An end tag is not allowed - it is an EMPTY element.

Required Attribute(s):

FR=nnnn Consisting of a 4-digit sequence number within a page.

HE=nnn Height: 3-digit expression in millimetres.

WI=nnn Width: 3-digit expression in millimetres.

Optional Attribute(s):

LX=nnnn 4-digit X-coordinate expressed in 1/10 millimetres referencing to the top left corner of the page.

LY=nnnn 4-digit Y-coordinate expressed in 1/10 millimetres referencing to the top left corner of the page.

FONT=name The font used in the text frame, eg Courier, Helvetica, etc.

SIZE=nn A 2 digit number for the point size of the font.

LS=n Where n is the number (may be decimal) of the line spacing within the text frame.

DTD Syntax:

```
<!ELEMENT txf    - o EMPTY                        -- Text frame               -->
<!ATTLIST txf  fr  NUTOKEN  #REQUIRED            -- Txf identity            -->
              he  NUMBER  #REQUIRED            -- Height in mm           -->
              wi  NUMBER  #REQUIRED            -- Width in mm            -->
              lx  NUMBER  #IMPLIED            -- X-coord 1/10 mm       -->
              ly  NUMBER  #IMPLIED            -- Y-coord 1/10 mm       -->
              font CDATA  #IMPLIED            -- Font name              -->
              size NUMBER  #IMPLIED            -- Font point size        -->
              ls  NUTOKEN #IMPLIED            -- Line spacing           -->
```

Example:

```
<PATDOC CY=JP>
<SDOAB>
<TXF FR=0001 HE=080 WI=080 LX=0200 LY=1800>
<P>Japanese Patent Office abstract...
</SDOAB></PATDOC>
```

26. <DP> : Document Page

This indicates the beginning of a page. No end tag is necessary.

Note: The use of this tag is optional since it is a formatting tag. It may be discarded at the time of presentation. However, it may be useful for patent documents where page citation is common and may need to be preserved in electronic document systems.

Required Attribute(s):

N=nnnn 4-digit number being the page number per document.

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT dp    - o EMPTY                        -- Document page break    -->
<!ATTLIST dp    n  NMTOKEN  #REQUIRED            -- Document page number   -->
```

Example:

<DP N=6>This is the start of page 6

27. <PCL> : Page CoLumn

This indicates the beginning of a column in a page. It should always be preceded by <TXF> tag. No end tag is necessary.

Note: The use of this tag is optional since it is a formatting tag. It may be discarded at the time of presentation. However, it may be useful for patent documents, where column citation is used, and may need to be preserved in electronic document systems.

Required Attribute(s):

N=nnnn 4-digit number being the column number.

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT pcl      - o EMPTY                -- Page column      -->
<!ATTLIST pcl     n  NMTOKEN  #REQUIRED     -- Page column number -->
```

Example:

<PCL N=2>This is the start of column 2

28. <PLN> : Page LiNe

This indicates the beginning of a line within a page. It should always be preceded by a <TXF> tag. No end tag is necessary.

Note: The use of this tag is optional since it is a formatting tag. It may be discarded at the time of presentation. However, it may be useful for patent documents, where line number citation is common, and may need to be preserved in electronic document systems.

Required Attribute(s):

N=nnnn 4-digit number being the line number.

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT pln     - o EMPTY                -- page line        -->
<!ATTLIST pln    n  NMTOKEN  #REQUIRED     -- page line number -->
```

Example:

<PLN N=15>This is the start of line 15

Lists

TABLE OF SGML TAGS AND ATTRIBUTES

TAG	NAME	DESCRIPTION
<DD>	Definition Description	Indicates a text portion which is the description of a tagged item in a definition list. No end tag is necessary.
<DL>	Definition List	Indicates a text portion to be displayed as a list, each item comprising a term followed by a description. An end tag is required.
<DT>	Definition Term	Indicates a text portion which is the term in a definition list. No end tag is necessary.
	List Item	Indicates the beginning of an item which forms part of a simple, ordered or unordered list. No end tag is necessary.
	Ordered List	Indicates a text portion to be displayed as a list, each item being identified by a sequential number or letter. An end tag is required.
<SL>	Simple List	Indicates a text portion to be displayed as a simple list. An end tag is required.
	Unordered List	Indicates a text portion to be displayed as a list, each item identified by a symbol which is defined in a required attribute (ST). An end tag is required.

ATTRIBUTE	NAME	DESCRIPTION
COMPACT	COMPACT	Indicates if lists should be processed as compact
LEVEL	LEVEL	Nesting level of a list
NUMSTYLE	NUMSTYLE	Numeric style of a list
PREFIX	PREFIX	Prefix for each list item
ST	STyle	Style (character or graphic) of an unordered list item
TSIZE	Term SIZE	Indicates the horizontal space allowed for definition terms plus gutter space

SGML tags: description and usage

29. <DL> : Definition List

This indicates a text portion known as a definition or glossary list. A definition list contains one or more items, each followed by its description. The items are identified by the <DT> identifier and the description by the <DD> identifier. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

TSIZE= This attribute is used to specify the indent to be used for the definition description. It is normally larger than the maximum width of the terms.

COMPACT= Used to indicate when no blank lines are to be left between definition items at the time of presentation.

DTD Syntax:

```
<!ELEMENT dl    - - (dt,dd)+                -- Definition list          -->
<!ATTLIST dl    tsize      NUMBER #IMPLIED  -- Term size attribute      --
               compact (compact) #IMPLIED  -- Spacing between items    -->
```

Example: (see below)

30. **<DT> : Definition Term**

This indicates a term in a definition list. No end tag is necessary.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT dt    - o (%ptext;)              -- Definition term          -->
```

Example: (see below)

31. **<DD> : Definition Description**

This indicates the description of an item (term) marked <DT> in a definition list. No end tag is necessary.

Required Attribute(s):

None

Optional Attribute(s):

None

DTD Syntax:

```
<!ELEMENT dd    - o ((%ptext;)|p)+         -- Definition description   -->
```

Example: In this example it is assumed that none of the terms exceed the length that may have been specified as default for such lists.

```
EPO    European Patent Office    < DL>
      < DT>EPO
JPO    Japanese Patent Office    <DD>European Patent Office
      <DT>JPO
USPTO  United States Patent and Trademark Office <DD>Japanese Patent Office
      <DT>USPTO
      <DD>United States Patent and Trademark Office
      </DL>
```

32. : Ordered List

This indicates a portion of structured text known as a list. An ordered list will have a sequence of numbers or letters generated at the time the document is created, not at the time of presentation, to indicate the relative position in the list of each item. Lists may be nested. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

COMPACT= Used to indicate when no blank lines are to be left between items at the time of presentation.

LEVEL= Used to indicate the nesting level of a list.

NUMSTYLE= Used to indicate the numeric style of a list.

PREFIX= Used to indicate prefix for each list item.

DTD Syntax:

```
<!ELEMENT ol    - - (li)+                -- Ordered list                -->
<!ATTLIST ol    compact (compact) #IMPLIED -- Spacing between items      --
               level   NUMBER  #IMPLIED  -- Nesting level of list      --
               numstyle CDATA   #IMPLIED  -- Numbering style            --
               prefix   CDATA   #IMPLIED  -- Prefix for each list item  -->
```

Example: (see below)

33. <SL> : Simple List

This indicates a portion of structured text known as a list. A simple list will not have anything preceding the list items to indicate them as such. Lists may be nested. An end tag is required.

Required Attribute(s):

None

Optional Attribute(s):

COMPACT= Used to indicate when no blank lines are to be left between items at the time of presentation.

LEVEL= Used to indicate the nesting level of a list.

DTD Syntax:

```
<!ELEMENT sl      - - (li)+                -- Simple list                -->
<!ATTLIST sl     compact (compact) #IMPLIED -- Spacing between items     --
              level   NUMBER  #IMPLIED     -- Nesting level of list     -->
```

Example: (see below)

34. : Unordered List

This indicates a portion of structured text known as a list. An unordered list will have symbols generated at the time of presentation to indicate each item. Lists may be nested. An end tag is required.

Required Attribute(s):

ST= This attribute is followed by an identifier for the character or the graphic symbol required to indicate each separate item in the list.

Optional Attribute(s):

COMPACT= Used to indicate when no blank lines are to be left between items at the time of presentation.

LEVEL= Used to indicate the nesting level of a list.

DTD Syntax:

```
<!ELEMENT ul      - - (li)+                -- Unordered list                -->
<!ATTLIST ul     st          NMTOKEN  #REQUIRED     -- Ulist symbol                 --
              level   NUMBER  #IMPLIED     -- Nesting level of list     --
              compact (compact) #IMPLIED     -- Spacing between items     -->
```

Example: (see below)

