

-55-

the carer does not cease to be qualified for a carer pension merely because of that absence from Australia.

\$\$\$

<

5 "(1C) If, during a calendar year, the carer has accompanied the handicapped person outside Australia on more than one occasion, the carer ceases to be qualified for carer pension under subsection (1B) in that calendar year after he or she has, during that calendar year, qualified for carer pension under that subsection for periods that together add up to 3 months."

10 \$\$A

[C 05]Date of Commencement Field

This field is for the date an amendment or repeal of a specific Part, Section or regulation commenced. See also **[B 04]** for information on commencement.

15

EXAMPLE

Below is an example of a commencement provision from an amending ACT. Note how various sections of the amending ACT are allocated a date of commencement. Using the Section identified in TEXTBLOCK ID1 the date of commencement can be identified and entered in this field.

20

Thus if the TEXTBLOCK ID1 field were **ACT-19950105-SEC-8** then the date information required for this field would be 1 July 1995. See **(DD)** below.

25 SOCIAL SECURITY (NON-BUDGET MEASURES) LEGISLATION AMENDMENT ACT 1995 No. 105 of 1995 - SECT 2<

Commencement<

\$\$\$

30

\$\$NSECT

2.(1) Subject to this section, this Act commences on the day on which it receives the Royal Assent.

\$\$\$

35

<
(2) Subdivision A of Division 2 of Part 2 is taken to have commenced on 1 July 1993.

\$\$\$

-56-

<
(3) Sections 8 and 9 are taken to have commenced on **(DD)**1 July 1995.

\$\$\$

<

5 (4) Section 10 is taken to have commenced on 1 April 1993.

\$\$\$

[C 06]Date of Expiry Field

This field provides for amendments that are enacted for a period of time, for example; during a
10 special event such as the Olympics.

Note: Expiry provisions are rare and should only be added if clearly specified in the data as in the
example below.

15 **EXAMPLE**

See **(EE)** below.

OLYMPIC SECURITY (NON-BUDGET MEASURES) LEGISLATION AMENDMENT ACT 1997 No. 109 of
1995

20

\$\$NSECT

4. Section 298 of the Principal Act is amended for the period commencing at
midnight on 1 July 1997 and ending on midnight **(EE)**25 July 1997 by inserting
after subsection (2A):

25 \$\$\$

<

"(2B) Subject to subsection (2C), if:<

(a) a person (the 'carer') is personally providing constant care for a
severely handicapped person; and<

30 (b) the handicapped person is temporarily absent from Australia for a period
of not more than 3 months; and<

(c) the carer accompanies the handicapped person on his or her absence from
Australia;<

35 the carer does not cease to be qualified for a carer pension merely because of
that absence from Australia.

\$\$\$

<

-58-

TAMZ
APPENDIX D

Keying Guide for Australian Legislation Documents

General

Validation

All files produced must be parsed against the relevant DTD and each table should be viewed using a suitable Table renderer to make sure that they have been coded correctly.

DTD modifications

No local modifications must be made to the supplied DTD's. If there are any situations in which it is thought that a change to a DTD is required then the requested change and the reasons for it must be submitted to SGMLSE. If a change is deemed necessary, then the DTD will be changed in the UK and resupplied. This is necessary to maintain consistency in the DTD's being used at both ends in the process.

Files

The SGML files that will be supplied have already been partially processed but will be invalid according to the DTD (especially the tables).

Each file should omit the document type declaration and begin directly with the root element.

Carriage return characters

Carriage return characters must not appear in any element which has #PCDATA within its content model. If it is wished to use carriage return characters to shorten line lengths, then they must be placed in positions where they will be ignored by an SGML parser e.g. in places where #PCDATA is not allowed, or within start and end tags in places where separator characters are allowed.

Case

Element and attribute names are case-insensitive. They may be entered in either uppercase, lowercase or a mixture.

Attribute values are usually case-insensitive. The only time that they are case-sensitive is when they have a declared type of CDATA, in which case the string values should be entered directly as they appear in the text.

-59-

Markup minimisation

No non-empty elements have omissible start or end tags, but the empty end tag `</>` can be used to end the currently open element. A carriage return character can not occur within an empty end tag.

5 **DTD structure**

There are four DTDs and a common element declaration that are used to define the structure of the legislation.

REGS.DTD

10 This DTD has the public identifier "`-//SGMLSE//DTD 1.0 Regulations//EN`" and contains the declaration for the regulations. It has two parameter entity references which include "`-//SGMLSE//DTD 1.0 Act//EN`" and "`-//SGMLSE//DTD 1.0 Regulation//EN`".

The file *regs.sgm* produced by the Perl script *regs.pl* conforms to *regs.dtd* and calls in all acts as parameter entities.

ACTS.DTD

15 This DTD has the public identifier "`-//SGMLSE//DTD 1.0 Acts//EN`" and contains the declaration for the acts. It has a parameter entity references which includes "`-//SGMLSE//DTD 1.0 Act//EN`".

The file *acts.sgm* produced by the Perl script *acts.pl* conforms to *acts.dtd* and calls in all acts as parameter entities.

20 **ACT.DTD**

This DTD has the public identifier "`-//SGMLSE//DTD 1.0 Act//EN`" and contains the declaration of an act.

REG.DTD

25 This DTD has the public identifier "`-//SGMLSE//DTD 1.0 Regulation//EN`" and contains the declaration of a regulation.

COMMON.ELT

This list of elements has the public identifier "`-//SGMLSE//ELEMENTS 1.0 Common Elements//EN`" and includes element and entity definitions common to all document types.

30 **Character entities**

The character entities allowed have been selected from the ISO public sets *isogr1*, *isogr3*, *isolat1*, *isonum*, *isopub* and *isotech*.

These characters are translated one-to-one to the equivalent character in the Times New Roman True Type font.

35 The `<quote>` or `<blockquote>` elements should be used instead to surround any quoted text.

Use Quotes (") at the beginning and at the end of these elements.

-60-

Labels and identifiers

The major structural elements - act, reg, part, schedule, section, sub, s-sub, ss-sub, ... all have a required label attribute (lbl). Unique identifiers should be generated for these elements. The label for these elements is the preceding number or letter WITHOUT any punctuation or parentheses. For example :

1979 No. 141 The charge to income tax - REG 1

Income Tax

1. Fred ...

10 1. (1) This is ...

```
<section type="reg" lbl="1" ID="CWACT-19790141-SEC-1">
```

```
<title>Income Tax</>
```

```
<list>
```

```
<li lbl="1"><p>Fred ...</p>
```

15 </list>

```
<li lbl="1"><p>This is ...</p>
```

```
</li>
```

```
</list>
```

20 </section>

Convert list and li items back to the original look in Folio Views, ie to

1. Fred ...

1. (1) This is ...

Cross references

25 All cross references point directly to a target by providing the id of the target as the value of an attribute of the xref element. For details of the format of cross reference identifier strings, see the description of the xref element below.

All ID's are marked unchanged as Jump Destinations (JD's).

```
<SECTION ID="CWACT-19950104-SEC-1" LBL="1">
```

30 *becomes:*

```
<JD:"="CWACT-19950104-SEC-1">
```

REGS DTD ELEMENT

REGS

35 This is the root element of the Regulations. Its definition is :

```
<!ELEMENT regs - - (title, header?, reg+) >
```

That is, it contains a required title element, followed by an optional header element followed by 1 or more act element.

Not translated

ACTS DTD ELEMENT

ACTS

5 This is the root element of the Acts. Its definition is :

```
<!ELEMENT acts - - (title, header?, act+) >
```

That is, it contains a required title element, followed by an optional header element followed by 1 or more act element.

Not translated

10 **REG DTD ELEMENTS**

REG

This is the root element of the dtd for a Regulation. Its definition is :

```
<!ELEMENT reg - - (title, notes?, provisions?, (preamble|long-
title)?,
15 ((order+|(section|schedule)+|chapter+|part+), schedule*)) >
<!ATTLIST reg id ID #REQUIRED
        lbl CDATA #REQUIRED
        insert-date NUMBER #IMPLIED
        insert-leg IDREF #IMPLIED
20 repeal-date NUMBER #IMPLIED
        repeal-leg IDREF #IMPLIED
        amend-date NUMBER #IMPLIED
        amend-leg IDREF #IMPLIED
>
```

25 The insert-date attribute should be used to insert the date YYYYMMDD that the REG was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the REG.

The repeal-date attribute should be used to insert the date YYYYMMDD that the REG was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the REG.

30 The amend-date attribute should be used to insert the date YYYYMMDD that the REG was amended. The amend-leg should be used to insert the ID of the legislation that amended the REG.

35 *The title gets marked up as Level "Heading Level 1". The period between the insert-date and the repeal-date or amend-date (that is the period during which the above element was in force) gets marked up as follows:*

The Folio Views markup used is Groups. See the manual for explanations. We will use three kinds of groups: Inforce-yyyy, Inforce-yyyy-mm and Inforce-yyyy-mm-dd (yyyy can

be any year, mm can be any month from 1 - 12 and dd can be any day from 1 - 31). If the element was valid throughout a year yyyy, the element becomes a member of the Inforce-yyyy group. If the element was valid only for some months within a year, it becomes a member of the relevant Inforce-yyyy-mm groups. If the element was only valid for some days within a month, the element becomes a member of the relevant Inforce-yyyy-mm-dd groups. Example:

If the element was valid from 1/7/94 to 7/4/96, then the element belongs to the following groups:

Inforce-94-07, Inforce-94-08, Inforce-94-09, Inforce-94-10, Inforce-94-11, Inforce-94-12, Inforce-95, Inforce-96-01, Inforce-96-02, Inforce-96-03, Inforce-96-04-01, Inforce-96-04-02, Inforce-96-04-03, Inforce-96-04-04, Inforce-96-04-05, Inforce-96-04-06 and Inforce-96-04-07.

Note that the element is not part of the Inforce-94 group because the element wasn't in force throughout of 1994. Nor is the element part of the Inforce-96-04 group because the element wasn't in force throughout April 1996.

If the enduser wants to search for all elements that are valid as of a particular date then the enduser can enter that date in a Query Template in the form DD/MM/YYYY. The Query Template then searches the Folio Views infobase for all elements that belong to the groups Inforce-yyyy, Inforce-yyyy-mm and Inforce-yyyy-mm-dd.

ACT DTD ELEMENTS

ACT

This is the root element of the dtd for an Act. Its definition is :

```
<!ELEMENT act - - (title, notes?, provisions?, (preamble|long-
title)?, (section+|chapter+|part+|sub+)*, schedule*)>
<!ATTLIST
    act id ID #REQUIRED
        date CDATA #IMPLIED
        insert-date NUMBER #IMPLIED
        insert-leg IDREF #IMPLIED
        repeal-date NUMBER #IMPLIED
        repeal-leg IDREF #IMPLIED
        amend-date NUMBER #IMPLIED
        amend-leg IDREF #IMPLIED
>
```

That is, it contains a required title element, followed by optional notes, provisions, and preamble and either one or more reg, order, rule, section, chapter or part elements. It has a required id attribute and implied date attribute

366274US

-63-

The insert-date attribute should be used to insert the date YYYYMMDD that the ACT was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the ACT.

5 The repeal-date attribute should be used to insert the date YYYYMMDD that the ACT was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the ACT.

The amend-date attribute should be used to insert the date YYYYMMDD that the ACT was amended. The amend-leg should be used to insert the ID of the legislation that amended the ACT.

10 **Examples**

1979 No. 141 The charge to income tax

```
<act lbl="141" ID="CWACT-19790141-TXT-0">
<title>1979 No. 141 The charge to income tax</>
```

1979 No. 141 The charge to income tax*1*

```
15 <act lbl="141" ID="CWACT-19790141-TXT-0">
<title>1979 No. 141 The charge to income tax<noteref ref="CWACT-
19790141-note-1"></>
```

The noteref gets marked up as a popup link. The text of the note becomes the text within the popup link.

20 *Treated the same as Reg DTD Elements. See there for details.*

COMMON ELEMENTS

PART

```
<!ELEMENT part - - (title, (%plevel;)*,
25 (chapter+|division+|section+)*, notes?)
<!ATTLIST partid ID #REQUIRED
lbl CDATA #IMPLIED
insert-date NUMBER #IMPLIED
insert-leg IDREF #IMPLIED
30 repeal-date NUMBER #IMPLIED
repeal-leg IDREF #IMPLIED
amend-date NUMBER #IMPLIED
amend-leg IDREF #IMPLIED
>
```

35 The insert-date attribute should be used to insert the date YYYYMMDD that the PART was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the PART.

-64-

The repeal-date attribute should be used to insert the date YYYYMMDD that the PART was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the PART.

The amend-date attribute should be used to insert the date YYYYMMDD that the PART was amended. The amend-leg should be used to insert the ID of the legislation that amended the PART.

Example

```
<PART lbl="1" ID="CWACT-19790141-PT-1">
<title></title>
```

```
<p>
```

The title gets marked up as Level "Heading Level 2". Dates get treated the same way as in Regs DTD Elements. See there for details.

ORDER

```
<!ELEMENT order - - (title,
15 (division|section|part|schedule)+)>
<!ATTLIST   order   id ID #REQUIRED
              lbl CDATA #IMPLIED
              insert-date    NUMBER    #IMPLIED
              insert-leg     IDREF     #IMPLIED
20              repeal-date   NUMBER    #IMPLIED
              repeal-leg     IDREF     #IMPLIED
              amend-date     NUMBER    #IMPLIED
              amend-leg      IDREF     #IMPLIED
```

```
>
```

The insert-date attribute should be used to insert the date YYYYMMDD that the ORDER was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the ORDER.

The repeal-date attribute should be used to insert the date YYYYMMDD that the ORDER was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the ORDER.

The amend-date attribute should be used to insert the date YYYYMMDD that the ORDER was amended. The amend-leg should be used to insert the ID of the legislation that amended the ORDER.

Example

```
35 1979 No. 140 FEDERAL COURT RULES - ORDER 3<
```

```
ORDER 3<
```

```
TIME<
```

```
$$T
```

-65-

\$\$A

\$\$T

1979 No. 140 FEDERAL COURT RULES - RULE 1<

<ORDER lbl="3" ID="CWACT-19790140-ORD-3">

5 <title>TIME</title>

<section type="rule" lbl="1" ID="CWACT-19790140-ORD-3.1">

The title gets marked up as Level "Heading Level 2". Dates get treated the same way as in Regs DTD Elements. See there for details.

DIVISION

10 <!ELEMENT division - - (title, (sub-division+ | section+)) >

<!ATTLIST division id ID #REQUIRED

lbl CDATA #IMPLIED

insert-date NUMBER #IMPLIED

insert-leg IDREF #IMPLIED

15 repeal-date NUMBER #IMPLIED

repeal-leg IDREF #IMPLIED

amend-date NUMBER #IMPLIED

amend-leg IDREF #IMPLIED

>

20 The insert-date attribute should be used to insert the date YYYYMMDD that the DIVISION was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the DIVISION.

The repeal-date attribute should be used to insert the date YYYYMMDD that the DIVISION was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the DIVISION.

25 The amend-date attribute should be used to insert the date YYYYMMDD that the DIVISION was amended. The amend-leg should be used to insert the ID of the legislation that amended the DIVISION.

Example

30 1979 No. 140 FEDERAL COURT RULES - DIVISION 1<

Division 1-General<

\$\$T

\$\$A

\$\$T

35 1979 No. 140 FEDERAL COURT RULES - RULE 1<

Cases for service of originating process<

<DIVISION lbl="1" ID="CWACT-19790140-DIV-1>

<title>General</title>

19790140-ORD-3.1

-66-

```
<RULE lbl="1" ID="CWACT-19790140-DIV-1.1">
```

```
<title> Cases for service of originating process </title>
```

The title gets marked up as Level "Heading Level 3". Dates get treated the same way as in Regs DTD Elements. See there for details.

5 **SUB-DIVISION**

```
<!ELEMENT sub-division - - (title, section+) >
```

```
<!ATTLIST sub-division id ID #REQUIRED
```

```
lbl CDATA #IMPLIED
```

```
insert-date NUMBER #IMPLIED
```

10 insert-leg IDREF #IMPLIED

```
repeal-date NUMBER #IMPLIED
```

```
repeal-leg IDREF #IMPLIED
```

```
amend-date NUMBER #IMPLIED
```

```
amend-leg IDREF #IMPLIED
```

15 >

The insert-date attribute should be used to insert the date YYYYMMDD that the SUB-DIVISION was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the SUB-DIVISION.

20 The repeal-date attribute should be used to insert the date YYYYMMDD that the SUB-DIVISION was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the SUB-DIVISION.

The amend-date attribute should be used to insert the date YYYYMMDD that the SUB-DIVISION was amended. The amend-leg should be used to insert the ID of the legislation that amended the SUB-DIVISION.

25 *The title gets marked up with a Paragraph Style (see Folio Views Infobase Production Kit Manual for details) 'Subdivision'. Dates get treated the same way as in Regs DTD Elements. See there for details.*

PROVISIONS

```
<!ELEMENT provisions - - (title, tblblk) >
```

30 **Example**

```
$$TABLE
```

```
TABLE OF PROVISIONS<
```

```
$$P
```

```
Order<
```

35 \$\$P

```
1. Preliminary<
```

```
$$P
```

```
2. Sittings and Vacation<
```

-67-

```

<PROVISIONS>
<TITLE>TABLE OF PROVISIONS>
<TBLBLK>
. .
5 </TBLBLK>
</PROVISIONS>

```

Provisions get ignored. The Table of Provisions gets generated by the conversion program from the following Elements: Part, Chapter, Order, Division, Subdivision and Section. The generated Table of Provision is stored just before the first Part, Chapter, Order, Division, Subdivision and Section.

CHAPTER

```

<!ELEMENT chapter - - (title, (%plevel;)*,
(part+|section+|article+)?>
<!--
15      chapter id ID #REQUIRED
      lbl CDATA #IMPLIED
      insert-date    NUMBER    #IMPLIED
      insert-leg     IDREF     #IMPLIED
      repeal-date    NUMBER    #IMPLIED
      repeal-leg     IDREF     #IMPLIED
20      amend-date    NUMBER    #IMPLIED
      amend-leg      IDREF     #IMPLIED
-->

```

The insert-date attribute should be used to insert the date YYYYMMDD that the CHAPTER was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the CHAPTER.

The repeal-date attribute should be used to insert the date YYYYMMDD that the CHAPTER was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the CHAPTER.

The amend-date attribute should be used to insert the date YYYYMMDD that the CHAPTER was amended. The amend-leg should be used to insert the ID of the legislation that amended the CHAPTER.

The title gets marked up as Level "Heading Level 2". Dates get treated the same way as in Regs DTD Elements. See there for details.

SECTION

This is a sub element of a Regulation. Its definition is :

```

<!ELEMENT section - - (title, (%unstruct-cont;))>
<!--
35      section id ID #REQUIRED
-->

```

-68-

```

    lbl CDATA #IMPLIED
        insert-date    NUMBER    #IMPLIED
        insert-leg     IDREF     #IMPLIED
        repeal-date    NUMBER    #IMPLIED
5       repeal-leg     IDREF     #IMPLIED
        amend-date     NUMBER    #IMPLIED
        amend-leg      IDREF     #IMPLIED
    >

```

10 That is, it contains a required title element, followed by unstructured content. It has a required lbl attribute and a required id attribute.

The insert-date attribute should be used to insert the date YYYYMMDD that the SECTION was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the SECTION.

15 The repeal-date attribute should be used to insert the date YYYYMMDD that the SECTION was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the SECTION.

The amend-date attribute should be used to insert the date YYYYMMDD that the SECTION was amended. The amend-leg should be used to insert the ID of the legislation that amended the SECTION.

20 **Examples**

1979 No. 141 The charge to income tax - SECT. 1

Income Tax

1979 No. 141 The charge to income tax - REG 2

Tax

25 3. Income tax shall be charged...

```

<section lbl="1" id="CWACTION-1979141-SEC-1">
<title>Income tax</title>
<section type="reg" lbl="2" id="CWACTION-1979141-SEC-
1.2"><title>Tax</title>
30 <p>Income tax shall be charged...</p>

```

The title gets marked up as Level "Heading Level 4". Dates get treated the same way as in Regs DTD Elements. See there for details.

SCHEDULE

This is a sub element of a Regulation. Its definition is :

```

35 <!ELEMENT schedule - - (title?, (%unstruct-cont;))>
<!ATTLIST
    schedule id ID #REQUIRED
        lbl CDATA #IMPLIED
        insert-date    NUMBER    #IMPLIED

```

-69-

	insert-leg	IDREF	#IMPLIED
	repeal-date	NUMBER	#IMPLIED
	repeal-leg	IDREF	#IMPLIED
	amend-date	NUMBER	#IMPLIED
5	amend-leg	IDREF	#IMPLIED

>

That is, it contains a required title element, followed by unstructured content. It has a required lbl attribute and a required id attribute.

The insert-date attribute should be used to insert the date YYYYMMDD that the SCHEDULE was inserted. The insert-leg should be used to insert the ID of the legislation that inserted the SCHEDULE.

The repeal-date attribute should be used to insert the date YYYYMMDD that the SCHEDULE was repealed. The repeal-leg should be used to insert the ID of the legislation that repealed the SCHEDULE.

The amend-date attribute should be used to insert the date YYYYMMDD that the SCHEDULE was amended. The amend-leg should be used to insert the ID of the legislation that amended the SCHEDULE.

Examples

1979 No. 141 The charge to income tax - SCHEDULE 1

Income Tax

1. Income tax shall be charged...

```
<schedule lbl="1" id="CWACTION-1979141-SCH-1">
<title>Income tax</>
<p>Income tax shall be charged...</>
```

The title gets marked up as Level "Heading Level 2". Dates get treated the same way as in Regs DTD Elements. See there for details.

FORM

```
<!ELEMENT form - (title, formreg, front, back?) >
<!ATTLIST form %reqlbl;
%reqid;>
```

Each form has a title, regulation, a front and an optional back. Each has a required lbl attribute and a required ID.

Example

FORM A<

```
Regulation 7<
(Front of Form)<
COMMONWEALTH OF AUSTRALIA<
Trade Practices Act 1974-Sub-section 88 (1)<
```

-70-

EXCLUSIONARY PROVISIONS:<
APPLICATION FOR AUTHORIZATION<

To the Trade Practices Commission: Application is hereby made under sub-section 88 (1) of the Trade

5 (Back of Form)<

DIRECTIONS<

1. Where there is insufficient space on this form to furnish the required information, the information is to be shown on separate sheets, numbered consecutively and signed by or on behalf of the applicant.

10 <FORM lbl="a" ID="CWACT-19790141-FORM-A">

<FORMREG>Regulation 7</FORMREG>

<FRONT>

<ASIS>

COMMONWEALTH OF AUSTRALIA<

15 Trade Practices Act 1974-Sub-section 88 (1)<

EXCLUSIONARY PROVISIONS:<

APPLICATION FOR AUTHORIZATION<

To the Trade Practices Commission: Application is hereby made under sub-section 88 (1) of the Trade

20 </ASIS>

</FRONT>

<BACK>

<ASIS>

DIRECTIONS<

25 1. Where there is insufficient space on this form to furnish the required information, the information is to be shown on separate sheets, numbered consecutively and signed by or on behalf of the applicant.

</ASIS>

</BACK>

30 </FORM>

Forms get marked up with a Paragraph Style 'Forms'. The markup of 'Front' and 'Back' gets inserted as hidden text (hidden text is visible on the screen but doesn't show when the Form gets printed.

FORM ELEMENTS

35 **FORMREG**

<!ELEMENT formreg - - (#PCDATA) >

see above example

-71-

FRONT`<!ELEMENT front - - (asis) >`

see above example

BACK5 `<!ELEMENT back - - (asis) >`

see above example

ASIS`<!ELEMENT asis - - (#PCDATA) >`

date is displayed as-is

10 see above example

HEADER ELEMENTS

HEADER`<!ELEMENT header - - (scope?, updated?) >`**SCOPE**15 `<!ELEMENT scope - - (%text;) >`**UPDATED**`<!ELEMENT updated - - (%text;) >`**NOTE ELEMENTS**

NOTES20 `<!ELEMENT notes - O (note+) >`

List of notes at the start of an act

NOTE`<!ELEMENT note - - (%plevel;) >``<!ATTLIST note id ID #REQUIRED>`

25 A single note

Forms get marked up with a Paragraph Style 'Forms'. The markup of 'Front' and 'Back' gets inserted as hidden text (hidden text is visible on the screen but doesn't show when the Form gets printed.

STRUCTURAL ELEMENTS

30

TITLE

A generic title, which may occur in several different contexts.

*Covered in Acts DTD Elements, Regs DTD Elements and so on.***INLINE ELEMENTS**

35 **BOLD**Used to mark any inline text which is set in a bold face other than a title or a label. It may contain text or any inline elements other than `<bold>`.

It has no attributes.

-72-

ILEQN

An inline equation. This is a mathematical equation which is embedded in a line of text characters or other inline elements. See the attached description of equations for further details.

5 **ITAL**

Used to mark any inline text which is set in a italic face other than a title or a label. It may contain text or any inline elements other than <italic>.

It has no attributes.

QUOTE

10 A sequence of text characters or inline elements surrounded by single or double paired quotation marks. The quotation mark characters must not be entered as text characters or entity references as they will be generated automatically.

SUBSCR

A Subscript (inferior).

15 **SUPER**

A superscript (superior).

Brought over from SGML to Folio Views with no changes.

CROSS REFERENCES**NOTEREF**

20 A reference to a NOTE element normally used in a TITLE element

Implemented as a popup link.

TEMPREF

A reference to a piece of legislation where the ID is not known. The tempref element will converted to an XREF element at a later date (when the ID is known).

25 *Not converted.*

RNGREF

A cross reference to a sequential range of targets, e.g. see Sections 3 to 7.

It has two required attributes, startref and endref. Startref is the id of the first of the targets referenced and endref is the id of the last target referenced. For a description of id strings, see the description of the xref element.

30

Converted as a Query Link.

XREF

A cross reference to a single target. It has a single attribute, ref, which must contain the id string of the target of the reference. Ids are not being entered on elements during keying, but will be generated automatically from the lbl attribute of elements. However, for xrefs it is necessary to work out what the id string of the target will be. The format of id strings is described below :

35

Converted as a jumplink.

-73-

ID Strings

Cross references to sections of Acts and Act Schedules should be marked up using the xref element as described above.

- 5 Id strings are made up of four sequential fields separated by a "-" (dash) character as follows :

field1-field2-field3-field4

field 1 is the type of document which is being referenced. The current valid value is ACT. field2 is an abbreviated form of the year and number of the Act, e.g. 19880001 is 1988

- 10 No. 1.

field3 identifies the type of object being referenced. Valid values are :

ORD order

DIV division

SCH schedule in an Act

- 15 SEC section in an Act

CH chapter in an Act

PT part in an Act

NOTE a note

- 20 field4 is the identifier of the element being referenced, which is formed by concatenating the values of the lbl attributes of the referenced element and its ancestor elements, separated by a "." (point) character. e.g. 1 or 1.1 or 1.1.a or 1.1.a.iv

Examples

```
<section lbl="1"><!-- id is ACT-19790141-SEC-1 -->
```

```
<title>The charge to income tax.</>
```

- 25 <list>

```
<li lbl="1">
```

```
<p>Income tax shall be charged...</>
```

Implemented as jumpdestination.

- 30 **BLOCK LEVEL ELEMENTS**

DPEQN

A display equation. This is a mathematical equation which is set on one or more lines by itself. See the attached description of equations for further details.

It has no attributes.

- 35 It has no attributes.

LIST

A list of related lines of text which are not sub, s-sub or ss-sub elements.

-74-

LI

An item in a list. This is a single line of text within a list.

It has a single attribute, marker, which has allowed values of bullet, dash or none, with a default of none. If a marker character precedes the list item then the relevant value
 5 should be entered for the attribute. The marker character should not be entered as text. If any marker character other than a bullet or dash is found, contact SGMLSE for a change to the DTD.

P

A single line of text.

10 **BLOCKQUOTE**

A non-inline quote. The quotation mark characters must not be entered as text characters or entity references as they will be generated automatically

TBLBLK

A container element for a table which has a title. It contains a required title element
 15 followed by a single table.

It has no attributes.

The above elements get implemented either without conversion or using paragraph styles.

TABLE ELEMENTS**General**

20 Arbortext tables expressed in tagged ASCII form must follow this basic structure:

```

<table>
<rowrule>
for each row
{
25   <tablerow>
      <cellrule>
      for each column
      {
          <tablecell>text</tablecell>
30   <cellrule>
      }
      </tablerow>
      <rowrule>
}
35 </table>
  
```

-75-

TABLE

The <table> tag has three required attributes. They must be specified correctly or the table will not be handled properly.

ncols=NUMBER The number of columns in the table. This value MUST agree with the
5 number of columns expressed by the required cwl attribute.

wdm=(25|50|75|100) The numbers indicate the width of the table as a percentage of
the page width.

cwl=LIST where LIST is a list of integers each separated by a colon. Each integer
represents the relative width of a column.

10 **Example**

A four column table which is the full width of the page. The second and third columns are
twice the width of the first column, and the fourth column is three times the width of the
first :

```
<table ncols="4" wdm="100" cwl="1:2:2:3">
```

15 or, equivalently,

```
<table ncols="4" wdm="100" cwl="5:10:10:15">
```

CELLRULE

Empty element. Specifies a vertical rule. It has a single attribute, rty, which specifies the
type of rule. Valid values for rty are :

20 "." (point) for a blank rule,
 "-" (dash) for a single rule,
 "=" (equals) for a double rule,
 "+" (plus) for a bold rule.

Example25

```
<cellrule rty=".">
```

for a blank rule, or

```
<cellrule rty="-">
```

for a single rule.

ROWRULE

30 Empty element. Specifies a sequence of horizontal rules, one per cell in the row. It has a
single attribute, rtl, which is a colon-delimited list of rule type specifiers. There must be
one rule type specifier for each cell in the row. The valid specifiers are as for cellrule
above.

Example

35 For a four-column table

```
<rowrule rtl="-:.:.:.-">
```

would draw a horizontal rule above cells one and four.

-76-

TABLEROW

Specifies a row in the table. It has a single attribute, `hdr`, which specifies whether or not the row is a header row in a table which will be repeated over page breaks. The only valid value is "1" (one), which indicates that the row is a header row. An omitted value
 5 for `hdr` indicates that the row is not a header row. A value is only valid on the first `<tablerow>` in the table.

Example

```
<tablerow hdr="1">
```

indicates that the row is a header row (iff the `<tablerow>` is the first in the table, else it
 10 will generate an error).

```
<tablerow>
```

indicates that the row is not a header row if the `<tablerow>` is the first in the table.

TABLECELL

Indicates a cell in a row in a table. It has four optional attributes :

15 `chj=(b|l|r|c)` Horizontal justification for that cell.

`b` for both right and left justified,

`l` for flush left,

`r` for flush right,

`c` for centred.

20 Default: left justified

`cvj=(t|c|b)` Vertical justification for that cell.

`t` for top justified,

`c` for centred, or

25 `b` for bottom justified.

Default: top justified

`spn=INTEGER` For horizontally spanned columns. `VAL` is a whole number representing how many columns are spanned. Note that for horizontal spans, the text appears in the LEFTMOST cell in the span, and all other cells in the span should be void
 30 of text.

Default: 1

`vspn=INTEGER` For vertically spanned rows. `VAL` is a whole number representing how many rows are spanned. Note that for vertical spans, the text appears in the LOWEST cell in the span, and all other cells in the span should be void of text.

35 Default: 1

Arbortext table example

```
-----  

|foo      |      fum      |      |
```

-77-

			fee
spanned		ugh	

```

5
<tbl>
<table wdm="100" cwl="3:4:3">
<rowrule rtl=":-:-">
<tablerow hdr="1">
10 <cellrule rty="-">
<tablecell>foo</tablecell>
<cellrule rty="-">
<tablecell chj="c">fum</tablecell>
<cellrule rty="-">
15 <tablecell chj="r" cvj="b">fee</tablecell>
<cellrule rty="-">
</tablerow>
<rowrule rtl=":-:-">
<tablerow>
20 <cellrule rty="-">
<tablecell spn="2" chj="c">spanned</tablecell>
<cellrule rty="-">
<tablecell chj="c">ugh</tablecell>
<cellrule rty="-">
25 </tablerow>
<rowrule rtl=":-:-">
</table>
</tbl>

```

30 *Tables get converted to Microsoft Word tables and then converted into Folio Views.*

EQUATION ELEMENTS

General

This maths DTD is a subset of the Arbortext maths DTD, which itself is derived from the AAP maths DTD.

35 In maths mode, all spaces are ignored (except in a <phr> element as described below). Correct spacing is handled automatically.

All alphabetical characters and symbols are treated as variables and set in italic face, unless they occur within <phr>, <rm> or <rf> elements.

-78-

All numeric characters and operators are set in roman face, unless they occur within an <it> element.

Greek symbols should be entered using the <g> element rather than entity references. E.g. <g>a</g> produces alpha, <g>b</g> beta, etc. Any entity references for Greek characters which appear in equations will be flagged as errors by the parsing program.

B

Bold text in an equation.

DE

Denominator of a fraction.

F

Inline equation.

FD

Display equation.

FEN

Fence. A pair of bracketed delimiters. The attribute lp (left post) defines the type of the left delimiter as below, and the following element rp (right post) defines the type of the right delimiter.

```
<!ATTLIST fen  lp          (par|sqb|cub|ang|vb)      vb  --
                par  left parenthesis      (
20                sqb  left square bracket   [
                cub   left curly brace      {
                ang   left angle bracket    <
                vb    left vertical bar     |
                -->
```

FR

Fraction.

G

Greek character or characters. Valid characters are :

	char	equivalent entity
30	a	alpha
	b	beta
	c	chi
	d	delta
	D	Delta
35	e	epsilon
	3	epsiv
	4	phiv
	f	phis

	F	Phi
	g	gamma
	G	Gamma
	h	eta
5	i	iota
	j	thetav
	k	kappa
	l	lambda
	L	Lambda
10	m	mu
	n	nu
	p	pi
	2	piv
	P	Pi
15	q	thetas
	Q	Theta
	r	rho
	s	sigma
	S	Sigma
20	9	rhov
	t	tau
	u	epsilon
	U	Upsilon
	v	sigmav
25	w	omega
	W	Omega
	x	xi
	X	Xi
	y	psi
30	Y	Psi
	z	zeta

INF

Inferior. Subscript in an equation.

IT

35 Italic text in an equation.

NU

Numerator of a fraction.

-80-

OVL

Overline.

PHR

5 Phrase. In a phrase all characters are set in roman face and keyed space characters are preserved. A phrase is essentially a temporary escape out of maths mode back into normal text mode.

RAD

10 Radical or root. Contains a radicand (<rcd>), which is the constructs which appear beneath the top horizontal bar, and an optional radix (rdx), which is the power of the root (e.g. square, cube, 4, etc.).

RCD

Radicand. The content of a root construct.

RDX

Radix. The power of a root.

15 **RF**

Roman Function. A function name set in roman face, such as log, sin, cos, lim, arg, etc. It differs from the <rm> element in that preceding and following space characters are generated to separate it from surrounding characters.

RM

20 Roman face. Used to force an alpha character to be displayed in normal face rather than be treated as a variable and displayed in italic face.

RP

25 Right delimiter of a fence. It has a single attribute, post, which defines the type of the delimiter. Valid values are the same as for the lp attribute of the <fen> element, except that they specify the right hand match for the relevant left post.

SUP

Superior. A superscript in an equation.

UNL

Underline.

30 **Arbortext equation examples**

... by multiplying by the fraction -

A

B

where -

35 <p>... by multiplying by the fraction -</>
 <dpeqn><fr><nu><rm>A</></><de><rm>B</></></></>
 <p>where -</p>

-81-

... by the formula -

$$\frac{\text{gross taxable income}}{\text{net assets}}$$

5 <p>... by the formula -</p>

```
<math display="block">\frac{\text{gross taxable income}}{\text{net assets}}</math>
```

10 *Equations get converted to Microsoft Word equations and then converted into Folio Views. Alternatively equations get converted to images and added to Folio Views as images.*

366274US 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22 23 24 25 26 27 28 29 30 31 32 33 34 35 36 37 38 39 40 41 42 43 44 45 46 47 48 49 50 51 52 53 54 55 56 57 58 59 60 61 62 63 64 65 66 67 68 69 70 71 72 73 74 75 76 77 78 79 80 81 82 83 84 85 86 87 88 89 90 91 92 93 94 95 96 97 98 99 100

-82-

APPENDIX E

```

<!SGML "ISO 8879:1986"
--
ArborText's default SGML declaration, modified to allow
5 longer id/idref's, and to use a number of special characters
within them.
--

CHARSET
10 BASESET "ISO 646-1983//CHARSET
International Reference Version (IRV)//ESC 2/5 4/0"
DESCSET
          0 9 UNUSED
          9 2 9
15 11 2 UNUSED
13 1 13
14 18 UNUSED
32 95 32
127 1 UNUSED
20 128 128 "High-order characters"

CAPACITY SGMLREF
TOTALCAP 200000
ENTCAP 35000
25 ENTCHCAP 35000
ELEMCAP 35000
GRPCAP 150000
EXGRPCAP 35000
EXNMCAP 35000
30 ATTCAP 50000
ATTCHCAP 35000
AVGRPCAP 35000
NOTCAP 35000
NOTCHCAP 35000
35 IDCAP 35000
IDREFCAP 35000
MAPCAP 35000
LKSETCAP 35000
LKNMCAP 35000
40

SCOPE DOCUMENT

SYNTAX
SHUNCHAR 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17
45 18 19 20 21 22 23 24 25 26 27 28 29 30 31 127
BASESET "ISO 646-1983//CHARSET
International Reference Version (IRV)//ESC 2/5 4/0"
DESCSET 0 128 0

```

-83-

```

128 128 "High-order characters"
FUNCTION RE 13
  RS 10
  SPACE 32
5  TABSEPCHAR 9
  NAMING LCNMSTRT ""
    UCNMSTRT ""
    LCNMCHAR "-./[]"
    UCNMCHAR "-./[]"
10  NAMECASE GENERAL YES
    ENTITY NO
  DELIM GENERAL SGMLREF
    SHORTREF SGMLREF
  NAMES SGMLREF
15  QUANTITY SGMLREF
    ATTCNT 100
    ATTSPLEN 960
    BSEQLEN 960
    DTAGLEN 16
20  DTEMPLN 16
    ENTLVL 16
    GRPCNT 100
    GRPGTCNT 96
    GRPLVL 16
25  LITLEN 800
    NAMELEN 64
    NORMSEP 2
    PILEN 1024
    TAGLEN 960
30  TAGLVL 24

FEATURES
MINIMIZE DATATAG NO OMITTAG YES RANK NO SHORTTAG
YES
35  LINK SIMPLE NO IMPLICIT NO EXPLICIT NO
    OTHER CONCUR NO SUBDOC NO FORMAL YES
    APPINFO NONE >

```

```

40  <!DOCTYPE legislation
    [
    <!--

```

```

45  DTD for Scantext/Abha Legislation - (C) Turn-Key Systems 1997

```

History:

1997-10-13: fix hist at beginning of regs

-84-

1997-10-07: restore <target> tag
 1997-10-02: enhanced support for regulations (hist, unconv etc)

```

-->
5
<!-- useful characters such as &mdash; -->
<!ENTITY % ISOpub public "ISO 8879-1986//ENTITIES Publishing//EN" >
%ISOpub;

10
<!-- PARAMETER ENTITIES USED TO SIMPLIFY DTD MARKUP -->
<!entity % major
"schedule|intcon|article|annex|clause|chapter|part|division|subdivn|section|subsec|
reg|subreg"
15
-- major levels -->

<!entity % minor "defn|para|subpar1|subpar2|subpar3|subpar4|point"
-- minor levels -->

20
<!entity % secreg "section | reg"
-- used where either sections or regs are appropriate -->

<!entity % level "%major;| %minor;"
-- all levels -->

25
<!entity % refs "term|l.ref|h.ref"
-- references which can be found in normal text -->

<!entity % raw "(rawtext|rawtable|unconverted)*"
30
-- material such as forms which remains as raw text -->

<!entity % effect "sc"
-- typographical effects (more to be added later) -->

35
<!entity % text "(%refs;|%effect;|#PCDATA)*"
-- normal text (including refs defined above) -->

<!entity % hnote "hist | note"
-- hist and note are temporarily interchangeable -->

40
<!entity % body "(p+|repealed), (%hnote;)*"
-- body of a legislative element -->

<!entity % lev.id "((label, desc?) | (desc, label?))"
45
-- reversible level id -->

<!-- MASTER ELEMENT -->

```

-85-

```

<!element legislation -- (act | regulations)+
  -- Acts and Regs are currently supported -->

5 <!-- DEFINITIONS OF ACTS AND CONSTITUENT ELEMENTS -->
<!element act -- (desc, (%hnote; | %raw;)*,
  longtitle, preamble?,
  (chapter+ | part+ | section+),
  schedule*, hist*)>
10 <!element longtitle -- (#PCDATA)>
<!element preamble -- (p+) +(para)>

<!element regulations -- (desc, (%hnote; | %raw;)*,
  (chapter+ | part+ | reg+),
  schedule*, hist*)>
15

<!-- MAJOR LEVELS -->
<!element chapter -- (label, desc, (%hnote;)*,
  (((%secreg;)*, part*) | article+))>
20 <!element part -- (label?, desc, hist*,
  (((%secreg;)*, division*) | (article+ | clause+)))>
<!element division -- (label?, desc, (p | %hnote;)*,
  (((%secreg;)*, subdivn*) | clause+))>
<!element subdivn -- (label?, desc, (%hnote;)*,
  ((%secreg;)+ | clause+))>
25 <!element section -- (%lev.id;,
  (subsec+, (modules | module+)? | repealed))>
<!element subsec -- (%lev.id;,
  ((p, (p | note)*, hist*) | (repealed, note*, hist*)))>
30 <!element modules -- (%lev.id;,
  module+)>
<!element module -- (%lev.id;,
  (submod+ | repealed))>
<!element submod -- (%lev.id;,
  ((p, (p | %hnote; | method | tabloid)* | (repealed, (%hnote;)*)))>
35 <!element reg -- (label?, desc, hist?,
  (subreg+ | repealed))>
<!element subreg -- (label, desc?, hist?,
  ((p, (p | note)*, hist*) | (repealed, note*, hist*)))>
40

<!-- SCHEDULES, CONVENTIONS etc -->
<!element schedule -- (label?, desc?, hist*,
  (chapter+ | part+ | section+ | item+ | p+ | tabloid+ | intcon)?,
  %raw;)>
45 <!element intcon -- (desc, preamble,
  (part+ | division+ | subdivn+ | article+), annex*)>
<!element annex -- (label?, desc, p*,
  (part* | division* | clause* | para*))>

```

-86-

```

5 <!element article -- (label, desc, p*,
    (division* | clause* | para*))>
  <!element clause -- (((label, desc?) | (desc, label)), p+)>
  <!element item -- (label?, desc?, %body;)>

  <!-- DEFINITIONS AND MINOR LEVELS -->
  <!element defn -- (%body;)>
  <!element note -- (label?, p+)>
10 <!element hist -- (label?, (p+ | %raw;))>
  <!element point -- (p+)>

  <!element para -- (label?, %body;)>
  <!element subpar1 -- (label?, %body;)>
15 <!element subpar2 -- (label?, %body;)>
  <!element subpar3 -- (label?, %body;)>
  <!element subpar4 -- (label?, %body;)>

  <!element method -- (%lev.id;, (p | %hnote; | step)+)>
20 <!element step -- (label, %body;)>

  <!-- COMMON CONSTITUENT ELEMENTS -->
  <!element label -- (%text;)
25 -- chapter number, section number, etc. -->
  <!element desc -- (%text;)
  -- chapter name, section name, etc. -->
  <!element term -- (%text;)
  -- defined term -->
30 <!element repealed - o EMPTY
  -- indicates that the enclosing level has been repealed -->
  <!element p -- (%minor| %refs;| %effect| tabloid| amend| target| #PCDATA)*
  -- textual paragraph at any level -->

35 <!-- UNCONVERTED MATERIAL -->
  <!element unconverted -- CDATA
  -- yet to be converted (eg. complex schedules) -->
  <!element rawtext -- CDATA
40 -- unconverted text (eg. forms) -->
  <!element rawtable -- CDATA
  -- unconverted tables (eg. amended provisions) -->

45 <!-- CROSS-REFERENCES -->
  <!element l.ref -- (%text;) -- legislation ref -->
  <!element h.ref -- (%text;) -- history ref -->

```


-87-

```

<!-- AMENDMENT MARKUP -->
<!element amend -- (%refs|%effect|quote|#PCDATA)*
    -- amendment = action + text -->
<!element target -- (#PCDATA)
5    -- target of amendment -->
<!element quote -- (label|desc|p| %level| %refs| %effect|#PCDATA)*
    -- quoted material -->

<!-- PSEUDO-TABLES -->
10 <!element tabloid -- (label?, desc?, (head | row)+)>
<!element head -- (cell+)>
<!element row -- (cell+)>
<!element cell -- (#PCDATA|p)*>

15 <!-- EFFECTS -->
<!element sc -- (#PCDATA) -- small caps -->

20 <!-- ATTRIBUTE LISTS -->
<!attlist act juris (cth|nsw|vic|qld|sa|wa|tas|act|nt|imp)
    #REQUIRED -- jurisdiction --
    year CDATA #REQUIRED -- year assented --
    number CDATA #REQUIRED -- act number --
25    reps CDATA #IMPLIED -- 2nd reading (HR) --
    senate CDATA #IMPLIED -- 2nd reading (Sen) --
    assent CDATA #IMPLIED -- assent date --
    cdate CDATA #IMPLIED -- commencement -->
<!attlist regulations juris (cth|nsw|vic|qld|sa|wa|tas|act|nt|imp)
30    #REQUIRED -- jurisdiction --
    year CDATA #REQUIRED -- year assented --
    number CDATA #REQUIRED -- act number --
    reps CDATA #IMPLIED -- 2nd reading (HR) --
    senate CDATA #IMPLIED -- 2nd reading (Sen) --
35    notified CDATA #IMPLIED -- notification date --
    cdate CDATA #IMPLIED -- commencement -->
<!attlist schedule id ID #IMPLIED -- legislation id --
    cdate CDATA #IMPLIED -- commencement --
    refsec CDATA #IMPLIED -- referring section -->
40 <!attlist chapter id ID #IMPLIED -- legislation id --
    cdate CDATA #IMPLIED -- commencement -->
<!attlist part id ID #IMPLIED -- legislation id --
    cdate CDATA #IMPLIED -- commencement -->
<!attlist division id ID #IMPLIED -- legislation id --
45    cdate CDATA #IMPLIED -- commencement -->
<!attlist subdivn id ID #IMPLIED -- legislation id --
    cdate CDATA #IMPLIED -- commencement -->
<!attlist section id ID #IMPLIED -- legislation id --

```


We claim:

1. A computer-implemented system for publishing an electronic publication using text-based data, comprising:
 - a plurality of predefined portions of text-based data with each predefined portion being stored;
 - at least one predefined portion being modified and stored;
 - a plurality of linking means of a markup language, each predefined portion of said text-based data and said at least one modified predefined portion of text-based data being encoded with at least one linking means; and
 - a plurality of attributes, each attribute being a point on an axis of a multidimensional space for organising said plurality of predefined portions and said at least one modified predefined portion of said text-based data.
2. The system according to claim 1, comprising means for searching within the system.
3. The system according to claim 2, wherein said searching means uses one or more attributes.
4. The system according to claim 2, wherein said searching means uses any predefined portion, any modification of a predefined portion, or any word or phrase within such predefined portion or such modification.
5. The system according to claim 1, further comprising means for searching at least one of said text-based predefined portions of said data using said plurality of attributes, wherein said plurality of attributes are coupled to each of said predefined portions by said respective linking means, and for retrieving one or more of said predefined portions using said plurality of attributes to define a point in said multidimensional space.
6. The system according to claim 1, wherein said markup language is Standard Generalised Markup Language (SGML) or eXtensible Markup Language (XML).
7. The system according to claim 6, wherein said text-based data is encoded using one or more Document Type Definitions (DTD) or Style Sheet Mechanisms (SSM).
8. The system according to claim 1, wherein said linking means comprises any piece of information additional to the body of the text-based data.
9. The system according to claim 8, wherein said linking means is a code or markup that allows departure and destination points to be created between portions of said text-based data.
10. The system according to claim 1, wherein said at least one linking means comprises an identification code for said respective predefined portion.
11. The system according to claim 1, wherein a first database comprises said plurality of predefined portions of text-based data.
12. The system according to claim 11, wherein a second database comprises said plurality of attributes for managing said first database.
13. The system according to claim 1, wherein said predefined portions are encoded with one or more attributes.
14. The system according to claim 1, wherein said respective predefined portion is changed by performing one of the group consisting of adding at least one attribute to said respective predefined portion, deleting at least one attribute from said respective predefined portion, and modifying at least one of the attributes of said respective predefined portion.
15. The system according to claim 1, wherein said respective predefined portion is changed by performing one of the group consisting of adding data to said respective predefined portion, deleting data from said respective predefined portion, and modifying data of said respective predefined portion.

16. The system according to claim 1, wherein said text-based data comprises legislation.
17. The system according to claim 16, wherein each of said plurality of predefined portions of said text-based data is a respective provision of said legislation.
18. The system according to claim 17, wherein said provision is a section or schedule of an Act, or a regulation or schedule of a Regulation(s).
19. The system according to claim 1, wherein each predefined portion is a block of said text-based data, said block being larger than a single word and less than an entire document of said text-based data.
20. A computer readable recording medium for publishing an electronic publication using text-based data, comprising:
 - a plurality of predefined portions of text-based data with each predefined portion being stored;
 - at least one predefined portion being modified and stored; and
 - a plurality of linking means of a markup language, each predefined portion of said text-based data and said at least one modified predefined portion of text-based data being encoded with at least one linking means; and
 - a plurality of attributes, each attribute being a point on an axis of a multidimensional space for organising said plurality of predefined portions and said at least one modified predefined portion of said text-based data.
21. The recording medium according to claim 20, wherein means for searching can be used to search the recording medium.
22. The recording medium according to claim 21, wherein said searching means uses one or more attributes.
23. The recording medium according to claim 21, wherein said searching means uses any predefined portion, any modification of a predefined portion, or any word or phrase within such predefined portion or such modification.
24. The recording medium according to claim 20, further comprising means for searching at least one of said predefined portions of said text-based data uses said plurality of attributes, wherein said plurality of attributes are coupled to each of said predefined portions by said respective linking means, and for retrieving one or more of said predefined portions using said plurality of attributes to define a point in said multidimensional space.
25. The recording medium according to claim 20, wherein said markup language is Standard Generalised Markup Language (SGML) or eXtensible Markup Language (XML).
26. The recording medium according to claim 25, wherein said text-based data is encoded using one or more Document Type Definitions (DTD) or Style Sheet Mechanisms (SSM).
27. The recording medium according to claim 20, wherein said linking means comprises any piece of information additional to the body of the text-based data.
28. The recording medium according to claim 27, wherein said linking means is a code or markup that allows departure and destination points to be created between portions of said text-based data.
29. The recording medium according to claim 20, wherein said at least one linking means comprises an identification code for said respective predefined portion.
30. The recording medium according to claim 20, wherein a first database comprises said plurality of predefined portions of said text-based data.
31. The recording medium according to claim 30, wherein a second database comprises said plurality of attributes for managing said first database.
32. The recording medium according to claim 20, wherein said predefined portions are encoded with one or more attributes.

33. The recording medium according to claim 20, wherein said respective predefined portion is changed by performing one of the group consisting of adding at least one attribute to said respective predefined portion, deleting at least one attribute from said respective predefined portion, and modifying at least one of the attributes of said respective predefined portion.

34. The recording medium according to claim 20, wherein said respective predefined portion is changed by performing one of the group consisting of adding data to said respective predefined portion, deleting data from said respective predefined portion, and modifying data of said respective predefined portion.

35. The recording medium according to claim 20, wherein said text-based data comprises legislation.

36. The recording medium according to claim 35, wherein each of said plurality of predefined portions of said text-based data is a respective provision of said legislation.

37. The recording medium according to claim 36, wherein said provision is a section or schedule of an Act, or a regulation or schedule of a Regulation(s).

38. The recording medium according to claim 20, wherein said recording medium is made from one of the group consisting of magnetic media, optical media, and magneto-optical media.

39. The recording medium according to claim 20, wherein each predefined portion is a block of said text-based data, said block being larger than a single word and less than an entire document of said text-based data.

40. A computer-implemented method for publishing an electronic publication using text-based data, comprising the steps of:

- providing a plurality of predefined portions of text-based data with each predefined portion being stored;
- encoding each predefined portion of said text-based data with at least one linking means of a markup language;
- providing at least one predefined portion being modified and stored;
- providing a plurality of attributes, each attribute being a point on an axis of a multidimensional space for organising said plurality of predefined portions and said at least one modified predefined portion of said text-based data.

41. The method according to claim 40, comprising the step of searching said text-based data.

42. The method according to claim 41, wherein said searching step uses one or more attributes.

43. The method according to claim 41, wherein said searching step uses any predefined portion, any modification of a predefined portion, or any word or phrase within such predefined portion or such modification.

44. The method according to claim 40, further comprising the step of searching at least one of said predefined portions of said text-based data using said plurality of attributes, wherein said plurality of attributes are coupled to each of

said predefined portions by said respective linking means, and for retrieving one or more of said predefined portions using said plurality of attributes to define a point in said multidimensional space.

45. The method according to claim 40, wherein said markup language is Standard Generalised Markup Language (SGML) or eXtensible Markup Language (XML).

46. The method according to claim 45, wherein said text-based data is encoded using one or more Document Type Definitions (DTD) or Style Sheet Mechanisms (SSM).

47. The method according to claim 40, wherein said linking means comprises any piece of information additional to the body of the text-based data.

48. The method according to claim 47 wherein said linking means is a code or markup that allows departure and destination points to be created between portions of said text-based data.

49. The method according to claim 40, wherein said at least one linking means comprises an identification code for said respective predefined portion.

50. The method according to claim 40, wherein a first database comprises said plurality of predefined portions of said text-based data.

51. The method according to claim 50, wherein a second database comprises said plurality of attributes for managing said first database.

52. The method according to claim 40, wherein said predefined portions are encoded with one or more attributes.

53. The method according to claim 40, wherein said respective predefined portion is changed by performing one of the group consisting of adding at least one attribute to said respective predefined portion, deleting at least one attribute from said respective predefined portion, and modifying at least one of the attributes of said respective predefined portion.

54. The method according to claim 40, wherein said respective predefined portion is changed by performing one of the group consisting of adding data to said respective predefined portion, deleting data from said respective predefined portion, and modifying data of said respective predefined portion.

55. The method according to claim 40, wherein said text-based data comprises legislation.

56. The method according to claim 55, wherein each of said plurality of predefined portions of text-based data is a respective provision of said legislation.

57. The method according to claim 56, wherein said provision is a section or schedule of an Act, or a regulation or schedule of a Regulation(s).

58. The method according to claim 40, wherein each predefined portion is a block of said text-based data, said block being larger than a single word and less than an entire document of said text-based data.

* * * * *