

## 9 Base Tag Set for Verse

This base tag set is intended for use when encoding texts which are entirely or predominantly in verse, and for which the elements for encoding verse structure already provided by the core tag set are inadequate.

The tags described in section 6.11.1 *Core Tags for Verse* include elements for the encoding of verse lines and line groups such as stanzas: these are available for any TEI document, irrespective of the base tag set it uses. Like the base tag sets for prose and for drama, the base tag set for verse additionally makes use of the tag set defined in chapter 7 *Default Text Structure* to define the basic formal structure of a text, in terms of <front>, <body> and <back> elements and the text-division elements into which these may be subdivided.

The base tag set for verse extends the facilities provided by these two tag sets in the following ways:

- “numbered” <lg> elements are provided, by analogy with the “numbered” *divn* class elements discussed in section 7.1 *Divisions of the Body* (see section 9.2 *Structural Divisions of Verse Texts*)
- a special purpose <caesura> element is provided, to allow for segmentation of the verse line (see section 9.3 *Components of the Verse Line*)
- a set of attributes is provided for the encoding of rhyme scheme and metrical information (see sections 9.4 *Rhyme and Metrical Analysis* and 9.5 *Rhyme*)

To enable the base tag set for verse texts, a parameter entity *TEI.verse* must be declared within the document type subset, the value of which is INCLUDE, as further described in section 3.3 *Invocation of the TEI DTD*. A document using this base tag set and no additional tag sets will thus begin as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE TEI.2 PUBLIC "-//TEI P4//DTD Main Document Type//EN" "tei2.dtd" [
  <!ENTITY % TEI.XML 'INCLUDE' >
  <!ENTITY % TEI.verse 'INCLUDE' >
]>
```

This declaration makes available the elements and attributes described in this chapter, in addition to those described in chapter 6 *Elements Available in All TEI Documents* and 7 *Default Text Structure*.

### 9.1 Structure of the Base Tag Set for Verse

The base tag set for verse contains the following entity declarations:

```
<!-- 9.1: Base Tag Set for Verse: entities-->
<!--Text Encoding Initiative Consortium:
Guidelines for Electronic Text Encoding and Interchange.
Document TEI P4, 2002.
Copyright (c) 2002 TEI Consortium. Permission to copy in any form
is granted, provided this notice is included in all copies.
These materials may not be altered; modifications to these DTDs should
be performed only as specified by the Guidelines, for example in the
chapter entitled 'Modifying the TEI DTD'
These materials are subject to revision by the TEI Consortium. Current versions
are available from the Consortium website at http://www.tei-c.org-->
<!--First, declare the class of
components specific to verse -->
<!ENTITY % x.comp.verse "" >
<!ENTITY % m.comp.verse "%x.comp.verse; %n.lg1; | %n.lg2; | %n.lg3; |
%n.lg4; | %n.lg5;">
<!ENTITY % mix.verse '| %m.comp.verse;' >
<!--Next define attributes common to metrical elements-->
<!ENTITY % a.metrical '
  met CDATA %INHERITED;
  real CDATA #IMPLIED
  rhyme CDATA #IMPLIED'>
<!ENTITY % a.enjamb '
  enjamb CDATA #IMPLIED'>
<!-- end of 9.1-->
```

The base tag set for verse contains the following element declarations:

```
<!-- 9.1: Base Tag Set for Verse-->
<!--First, declare the default text structure, which
is the same for verse as it is for other kinds of text. (But declare
it only if verse is the only base.)-->
<![%TEI.singleBase;[
<!ENTITY % TEI.structure.dtd PUBLIC "-//TEI P4//ELEMENTS Default Text
Structure//EN" 'teistr2.dtd' >
%TEI.structure.dtd;
]]>
<!--Finally, declare the elements specific to this base
tag set-->
<!--declarations from 9.2: Numbered line groups inserted here -->
<!--declarations from 9.3: Caesura inserted here -->
<!-- end of 9.1-->
```

## 9.2 Structural Divisions of Verse Texts

Like other kinds of text, texts written in verse may be of widely differing lengths and structures. A complete poem, no matter how short, may be treated as a free-standing text, and encoded in the same way as a distinct prose text. A group of poems functioning as a single unit may be encoded either as a <group> or as a <text>, depending on the encoder's view of the text. For further discussion, including an example encoding for a verse anthology, see chapter 7 *Default Text Structure*.

Many poems consist only of ungrouped lines. This short poem by Emily Dickinson is a simple case:

```
<text>
<front>
  <head>1755</head>
</front>
<body>
  <l>To make a prairie it takes a clover and one bee,</l>
  <l>One clover, and a bee,</l>
  <l>And revery.</l>
  <l>The revery alone will do,</l>
  <l>If bees are few.</l>
</body>
</text>
```

Often, however, lines are grouped, formally or informally, into stanzas, verse paragraphs, etc. The <lg> element defined in the core tag set (in section 6.11.1 *Core Tags for Verse*) may be used for all such groupings. It may thus serve for informal groupings of lines such as those of the following example from Allen Ginsberg:

```
<text>
<body>
  <head>My Alba</head>
  <lg type="free">
    <l>Now that I've wasted</l>
    <l>five years in Manhattan</l>
    <l>life decaying</l>
    <l>talent a blank</l>
  </lg>
  <lg>
    <l>talking disconnected</l>
    <l>patient and mental</l>
    <l>sliderule and number</l>
    <l>machine on a desk</l>
  </lg>
  <!-- ... -->
</body>
</text>
```

It may also be used to mark the verse paragraphs into which longer poems are often divided, as in the following example from Samuel Taylor Coleridge's *Frost at Midnight*:

```

<lg type="para">
  <l>The Frost performs its secret ministry,</l>
  <l>Unhelped by any wind. ...</l>
  <!-- ... -->
  <l>Whose puny flaps and freaks the idling Spirit</l>
  <l>By its own moods interprets, every where</l>
  <l>Echo or mirror seeking of itself,</l>
  <l part="I">And makes a toy of Thought.</l>
</lg>
<lg>
  <l part="F">But O! how oft,</l>
  <l>How oft, at school, with most believing mind</l>
  <l>Presageful, have I gazed upon the bars,</l>
  <l>To watch that fluttering <hi>stranger</hi>! ... </l>
  <!-- ... -->
</lg>
<lg>
  <l>Dear Babe, that sleepest cradled by my side,</l>
  <!-- ... -->
</lg>

```

Note, in the above example, the use of the `part` attribute on the `<l>` element, where a verse line is broken between two line groups, as discussed in section 6.11.1 *Core Tags for Verse*.

Most typically, however, the `<lg>` element is used to mark the highly regular line groups which characterize stanzaic and similar verse forms, as in the following example from Chaucer:

```

<lg>
  <l>Sire Thopas was a doghty swayn;</l>
  <l>White was his face as payndemayn,</l>
  <l>His lippes rede as rose;</l>
  <l>His rode is lyk scarlet in grayn,</l>
  <l>And I yow telle in good certayn,</l>
  <l>He hadde a semely nose.</l>
</lg>
<lg>
  <l>His heer, his ber was lyk saffroun,</l>
  <l>That to his girdele raughte adoun;</l>
  <!-- ... -->
</lg>

```

Like other text-division elements, `<lg>` elements may be nested hierarchically. For example, one particularly common English stanzaic form consists of a quatrain or sestet followed by a couplet. The `<lg>` element may be used to encode both the stanza and its components, as in the following example from Byron:

```

<lg type="stanza">
  <lg type="sestet">
    <l>In the first year of Freedom's second dawn</l>
    <l>Died George the Third; although no tyrant, one</l>
    <l>Who shielded tyrants, till each sense withdrawn</l>
    <l>Left him nor mental nor external sun:</l>
    <l>A better farmer ne'er brushed dew from lawn,</l>
    <l>A worse king never left a realm undone!</l>
  </lg>
  <lg type="couplet">
    <l>He died &mdash; but left his subjects still behind,</l>
    <l>One half as mad &mdash; and t'other no less blind.</l>
  </lg>
</lg>

```

Note the use of the `type` attribute to name the type of unit encoded by the `<lg>` element; this attribute is common to all members of the *divn* class (see section 7.1.1 *Un-numbered Divisions*).<sup>87</sup> “Sestet” and “couplet” might conceivably also be used as the values of the `met` attribute in a metrical analysis, for which see below, section 9.4 *Rhyme and Metrical Analysis*. The `type` attribute is intended solely for

<sup>87</sup> For discussion of other attributes of this class, see 7.1.4 *Partial and Composite Divisions*.

conventional names of different classes of text block; the `met` attribute is intended for systematic metrical analysis.

The above example uses ‘un-numbered’ line groups which can nest within each other to any depth. When the base tag set for verse is in use, ‘numbered’ line groups may also be used as an alternative.

`<lg1>` contains a first-level (i.e. largest) group of verse lines functioning as a formal unit e.g. a stanza, refrain, verse paragraph, etc.

`<lg2>` contains a second-level (i.e. second largest) group of verse lines functioning as a formal unit e.g. a stanza, refrain, verse paragraph, etc.

The base tag set for verse defines up to five such numbered line group elements, `<lg1>` to `<lg5>` inclusive. These function in exactly the same way as the numbered *divn* class elements discussed in section 7.1 *Divisions of the Body*.

As an example of their use, consider the Shakespearean sonnet. This may be divided into two parts: a concluding couplet, and a body of twelve lines, itself subdivided into three quatrains:

```
<text>
  <body>
    <lg1 type="body">
      <lg2 type="quatrain">
        <l>My Mistres eyes are nothing like the Sunne,</l>
        <l>Curra11 is farre more red, then her lips red</l>
        <l>If snow be white, why then her brests are dun:</l>
        <l>If haire be wiers, black wiers grown on her head:</l>
      </lg2>
      <lg2>
        <l>I have seene Roses damaskt, red and white,</l>
        <l>But no such Roses see I in her cheekes,</l>
        <l>And in some perfumes is there more delight,</l>
        <l>Then in the breath that from my Mistres reekes.</l>
      </lg2>
      <lg2>
        <l>I love to heare her speake, yet well I know,</l>
        <l>That Musicke hath a farre more pleasing sound:</l>
        <l>I graunt I never saw a goddesse goe,</l>
        <l>My Mistres when shee walkes treads on the ground.</l>
      </lg2>
    </lg1>
    <lg1 type="couplet">
      <l>And yet by heaven I think my love as rare,</l>
      <l>As any she beli'd with false compare.</l>
    </lg1>
  </body>
</text>
```

Particularly lengthy poetic texts are often subdivided into units larger than stanzas or paragraphs, which may themselves be subdivided. Spenser’s *Faery Queene*, for example, consists of twelve ‘books’ each of which contains a prologue followed by twelve ‘cantos’. Each prologue and each canto consists of nine-line ‘stanzas’, each of which follows the same regular pattern. Other examples in the same tradition are easy to find.

Large structures of this kind are most conveniently represented by the *divn* class elements such as `<div>` or `<div1>` described in section 7.1 *Divisions of the Body*. Thus the start of the *Faery Queene* might be encoded as follows:

```
<body>
  <div1 n="I" type="book">
    <div2 n="1" type="canto">
      <lg n="I.1.1" type="stanza">
        <l>A noble knight was pricking on the plain</l>
        <l>Ycladd in mightie armes and silver shielde,</l>
        <!-- ... -->
      </lg>
    </div2>
```

```

    </div1>
  </body>

```

The encoder must choose at which point in the hierarchy of structural units to introduce <lg> elements rather than a yet smaller <div> element: it would (for example) also be possible to encode the above example as follows:

```

<body>
  <div1 n="I" type="book">
    <div2 n="I.1" type="canto">
      <div3 n="I.1.1" type="stanza">
        <l>A noble knight was pricking on the plain</l>
        <l>Ycladd in mightie armes and silver shielde,</l>
        <!-- ... -->
      </div3>
    </div2>
  </div1>
</body>

```

One reason for preferring the former version is that not all of Spenser's stanzaic verse is organized into both cantos and books. In a corpus containing other works as well as the *Faery Queene*, it would be inconvenient for stanzas to appear in one part as <div3> elements, and in another as <div2> elements. Another way of avoiding this problem would be to use un-numbered <div> elements; see further 7.1 *Divisions of the Body*.

The numbered line group elements have the following formal definition:

```

<!-- 9.2: Numbered line groups-->
<!ELEMENT lg1 %om.R0;
((%m.Incl;)*, (head, (%m.Incl;)*)?, ((l|lg2), (%m.Incl;)*)+)>
<!ATTLIST lg1
  %a.global;
  %a.divn;
  TEIform CDATA 'lg1' >
<!ELEMENT lg2 %om.R0;
((%m.Incl;)*, (head, (%m.Incl;)*)?, ((l|lg3), (%m.Incl;)*)+)>
<!ATTLIST lg2
  %a.global;
  %a.divn;
  TEIform CDATA 'lg2' >
<!ELEMENT lg3 %om.R0; ((%m.Incl;)*, (head,
(%m.Incl;)*)?, ((l|lg4), (%m.Incl;)*)+)>
<!ATTLIST lg3
  %a.global;
  %a.divn;
  TEIform CDATA 'lg3' >
<!ELEMENT lg4 %om.R0; ((%m.Incl;)*, (head,
(%m.Incl;)*)?, ((l|lg5), (%m.Incl;)*)+)>
<!ATTLIST lg4
  %a.global;
  %a.divn;
  TEIform CDATA 'lg4' >
<!ELEMENT lg5 %om.R0; ((%m.Incl;)*, (head,
(%m.Incl;)*)?, (l, (%m.Incl;)*)+)>
<!ATTLIST lg5
  %a.global;
  %a.divn;
  TEIform CDATA 'lg5' >
<!-- end of 9.2-->

```

### 9.3 Components of the Verse Line

It is often convenient for various kinds of analysis to encode subdivisions of verse lines. The general purpose <seg> element defined in the tag set for segmentation and alignment (section 14.3 *Blocks, Segments and Anchors*) is provided for this purpose:

<seg> contains any arbitrary phrase-level unit of text (including other <seg> elements). Attributes include:

**subtype** provides a sub-categorization of the segment marked.

*Values* any string of characters.

To use this element together with the base tag set for verse, the tag set for segmentation and alignment must also be enabled, by an additional declaration in the document type subset, as further described in section 3.3 *Invocation of the TEI DTD*. A document using the base tag set for verse and this additional tag sets will thus begin as follows:

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE TEI.2 PUBLIC "-//TEI P4//DTD Main Document Type//EN" "tei2.dtd" [
<!ENTITY % TEI.XML 'INCLUDE' >
<!ENTITY % TEI.verse 'INCLUDE' >
<!ENTITY % TEI.linking 'INCLUDE' >
]>
```

In Old and Middle English alliterative verse, individual verse lines are typically split into half lines. The `<seg>` element may be used to mark these explicitly, as in the following example from Langland's *Piers Plowman*:

```
<l><seg>In a somer seson,</seg>
  <seg>whan softe was the sonne,</seg></l>
<l><seg>I shoop me into shroudes</seg>
  <seg>as I a sheep were,</seg></l>
<l><seg>In habite as an heremite </seg>
  <seg>unholy of werkes,</seg></l>
<l><seg>Went wide in this world </seg>
  <seg>wondres to here.</seg></l>
<!-- ... -->
```

The `<seg>` element can be nested hierarchically, in the same way as the `<lg>` element, down to whatever level of detailed structure is required. In the following example, the line has been divided into *feet*, each of which has been further subdivided into syllables.<sup>88</sup>

```
<l>
  <seg type="foot"
  ><seg type="syll">Ar</seg><seg type="syll">ma </seg><seg type="syll">vi</seg>
  </seg><seg type="foot"
  ><seg type="syll">rum</seg><seg type="syll">que </seg><seg type="syll">ca</seg>
  </seg><seg type="foot"
  ><seg type="syll">no </seg><seg type="syll">Tro</seg>
  </seg><seg type="foot"
  ><seg type="syll">iae </seg><seg type="syll">qui </seg>
  </seg><seg type="foot"
  ><seg type="syll">pri</seg><seg type="syll">mus </seg><seg type="syll">ab </seg>
  </seg><seg type="foot"
  ><seg type="syll">or</seg><seg type="syll">is </seg>
  </seg>
</l>
```

The `<seg>` element may be used to identify any subcomponent of a line which has content; its type attribute may characterize such units in any way appropriate to the needs of the encoder. For the specific case of labeling each foot with its formal type (“dactyl”, “spondee”, etc.), and each syllable with its metrical or prosodic status (syllables bearing primary or secondary stress, long syllables, short syllables), however, the specialized attributes `met` and `real` are defined, which provide a more systematic framework than the type attribute; see section 9.4 *Rhyme and Metrical Analysis* below.

In classical verse, a hexameter like that above may also be formally divided into two *cola* or ‘hemistiches’. This example provides a typical case, in that the boundary of the first colon falls in the middle of one of the feet (between the syllables “no” and “Tro”). If both kinds of segmentation are required, the `part` attribute might be used to mark the overlapping structure as follows.<sup>89</sup>

<sup>88</sup> The eccentric formatting of this example is a consequence of the fact that whitespace within the `<seg>` element is always retained: this is desirable for the lower level `<seg>` elements (those of type `syll`) but not for the higher level ones. Whitespace inside a start-tag, as here in the `foot` type `<seg>` elements, is always discarded and does not affect the result.

<sup>89</sup> For clarity of presentation, we have in this example adopted the convention that white space outside the syllable tags is to be ignored; as noted above, this is not generally enforceable in XML and this convention is not therefore recommended for normal use.

```

<l>
  <seg type="hemistich">
    <seg type="foot">
      <seg type="syll">Ar</seg>
      <seg type="syll">ma </seg>
      <seg type="syll">vi</seg>
    </seg>
    <seg type="foot">
      <seg type="syll">rum</seg>
      <seg type="syll">que </seg>
      <seg type="syll">ca</seg>
    </seg>
    <seg type="foot" part="I">
      <seg type="syll">no </seg>
    </seg>
  </seg>
  <seg type="hemistich">
    <seg type="foot" part="F">
      <seg type="syll">Tro</seg>
    </seg>
    <seg type="foot">
      <seg type="syll">iae </seg>
      <seg type="syll">qui </seg>
    </seg>
  </seg>
</l>

```

Instead of using the part attribute on the <seg> element, it might be simpler just to mark the point at which the caesura occurs. An additional element is provided for analyses of this kind, in which what is to be marked are points “between the words”, which have some significance within a verse line:

**<caesura>** marks the point at which a metrical line may be divided.

In classical prosody, the *caesura*, which occurs within a foot, is distinguished from a *diaeresis*, which occurs on a foot boundary (not to be confused with the division of a diphthong into two syllables, or the diacritic symbol used to indicate such division, each of which is also termed *diaeresis*). This distinction is rarely made nowadays, the term ‘caesura’ being used for any division irrespective of foot boundaries. No special-purpose <diaeresis> element is therefore provided.

As an example of the <caesura> element, we refer again to the example from Langland. An encoder might choose simply to record the location of the caesura within each line, rather than encoding each half-line as a segment in its own right, as follows:

```

<l>In a somer seson, <caesura/> whan softe was the sonne, </l>
<l>I shoop me into shroudes <caesura/> as I a sheep were, </l>
<l>In habite as an heremite <caesura/> unholy of werkes, </l>
<l>Went wide in this world <caesura/> wondres to here. </l>

```

Logically, the opposite of caesura might be considered to be enjambement. The base tag set for verse defines an enjamb attribute for the <l> element, which allows the presence or absence of enjambement to be signaled. The following lines demonstrate the use of the enjamb attribute to mark places where there is a discrepancy between the boundaries of the <l> elements and the syntactic structure of the verse (a discrepancy of some significance in some schools of verse):

```

<l enjamb="y">Un astrologue, un jour, se laissa choir</l>
<l>Au fond d'un puits.</l>

```

The elements discussed in this section are formally defined as follows:

```

<!-- 9.3: Caesura-->
<!ELEMENT caesura %om.RO; EMPTY>
<!ATTLIST caesura
  %a.global;
  TEIform CDATA 'caesura' >
<!-- end of 9.3-->

```

## 9.4 Rhyme and Metrical Analysis

When the base tag set for verse is in use, the following additional attributes are available to record information about rhyme and metrical form:

`met` contains a user-specified encoding for the conventional metrical structure of the element.

`real` contains a user-specified encoding for the actual realization of the conventional metrical structure applicable to the element.

`rhyme` specifies the rhyme scheme applicable to a group of verse lines.

These attributes may be attached to the `<l>` element, to any of its numbered equivalents `<l>`, `<l>`, etc., or to the higher-level text-division elements `<div>`, `<div1>`, etc. In general, the attributes should be specified at the highest level possible; they may not however be specifiable at the highest level if some of the subdivisions of a text are in prose and others in verse. All these attributes may also be attached to the `<l>` and `<seg>` elements, but the default notation for the rhyme attribute has no defined meaning when specified on `<l>` or `<seg>`. The value for these attributes may take any form desired by the encoder; the nature of the notation used will determine how well the attribute values can be processed by automatic means.

The primary function of the metrical attributes is to encode the conventional metrical or rhyming structure within which the poet is working, rather than the actual prosodic realization of each line; the latter can be recorded using the `real` attribute, as further discussed below. There is no provision at this time, however, for encoding the particular realization of a rhyme pattern.

### 9.4.1 Sample Metrical Analyses

As a simple example of the use of these attributes, consider the following lines from Pope's *Essay on Criticism*:

```
<div type='book' n='1' met='-+|-+|-+|-+|-' rhyme='aa'>
  <l> n='1' type="verse paragraph">
    <l>'Tis hard to say, if greater Want of Skill</l>
    <l>Appear in <hi>Writing</hi> or in <hi>Judging</hi> ill;</l>
    <l>But, of the two, less dang'rous is th'Offence,</l>
    <l>To tire our <hi>Patience</hi>, than mis-lead our <hi>Sense</hi>:</l>
  </l>
  <!-- ... -->
</div>
```

This text is written entirely in *heroic couplets*; each line is iambic pentameter (which, using a common notation, can be described with the formula `-+|-+|-+|-+|-`, each `-` denoting a metrically unstressed syllable, each `+` a metrically stressed one, each `|` a foot boundary, and the `/` a line-end), and the couplets rhyme (which can be represented with the conventional formula `aa`).

Because both rhyme pattern and metrical form are consistent throughout the poem, they may be conveniently specified on the `<div>` element; the values given for the attributes will be inherited by any metrical unit contained within the `<div>` elements of this poem, and must be interpreted in the appropriate way.

Since the notation used in the `met`, `real`, and `rhyme` attributes is user-defined, no binding description can be given of its details or of how its interpretation must proceed. (A default notation is provided for the rhyme attribute, which however the encoder can replace with another; see section 9.5 *Rhyme*.) It is expected, however, that software should be able to support these attributes in useful ways; the more intelligent the software is, and the more knowledge of metrics is built into it, the better it will be able to support these attributes. In the extract given above, for example, the `met` and `rhyme` attribute values specified on the `<div>` element are inherited directly by the `<l>` elements nested within it. Since the `met` value specifies the metrical form of a single verse line, the structure of the `<l>` as a whole is understood to involve as many repetitions of the pattern as there are lines in the verse paragraph. The same attribute value, when inherited in turn by the `<l>` element, must be understood *not* to repeat. With sufficiently sophisticated software, segments within the line might even be understood as inheriting precisely that portion of the formula which applies to the segment in question; this will, however, be easier to accomplish for some languages than for others.



The rhyme attribute in this example uses the default notation to specify a rhyme scheme applicable only to pairs of lines. As elsewhere, the default notation for the rhyme attribute has no meaning for metrical units at the line level or below. In verse forms where line-internal rhyme is structurally significant, e.g. in some skaldic poetry, the default notation is incapable of expressing the required information, since the rhyme pattern may need to be specified for units smaller than the line. In such cases, a user-specified rhyme notation must be substituted for the default notation, or else the rhyme pattern must be described using some alternative method (e.g. by using the <link> mechanism described below).

The precise semantics of the met attribute and the inferences which software is expected or able to draw from it, are implementation-dependent; so are the semantics and processing of the rhyme attribute, when user-specified notations are used.

A formal definition of the significance of each component of the pattern given as the value of the met attribute may be provided in the <metDecl> element within the <encodingDecl> element in the TEI header (see section 5.3.8 *The Metrical Declaration Element*). The encoder is free to invent any notation appropriate to his or her analytic needs, provided that it is adequately documented in this element. The notation may define metrical components using invented or traditional names (such as “iamb” or “hexameter”) or in terms of basic units such as codes for stressed or unstressed syllables, or a combination of the two.

The real (for “realization”) attribute may optionally be specified to indicate any deviation from the pattern defined by the met attribute which the encoder wishes to record. By default, the real attribute has the same value as the met attribute on the same element; it is only necessary to provide an explicit value when the realization differs in some way from the abstract metrical pattern. The tension between conventional metrical pattern and its realization may thus be recorded explicitly. For example, many readers of the above passage would stress the word “But” at the beginning of the third line rather than the word “of” following it, as the metrical pattern would normally require. This variation might be encoded as follows:

```
<l real='+ - | - + | - + | - + | - + '>But, of the two, ...</l>
```

Where the real attribute is used to over-ride the default or conventional metrical pattern, it applies only to the element on which it is specified. The default pattern for any subsequent lines is unaffected.

As it happens, this particular kind of variation is very common in the English iambic pentameter — it even has a name: *trochaic substitution* — an encoder might therefore choose to regard this not as an instance of a variant realization, but as an instance of a variant metrical form:

```
<l met='+ - | - + | - + | - + | - + '>But, of the two, ...</l>
```

Alternatively, a different metrical notation might be defined, in which this kind of variation was permitted throughout the text.

In choosing whether to over-ride a metrical specification in this way or by using the real attribute, the encoder is required to determine whether the change is a systematic or conventional one (as in this example) or an occasional variation, perhaps for local effect. In the following example, from Goethe’s *Auf dem See*, the variation is a matter of local realization:

```
<lgl type="chevy-chase-stanza"
  met="--+---+/-+---+"
  rhyme='ababcdcd'>
<l n='1'> Und frische Nahrung, neues Blut</l>
<l n='2' real="--+---+"> Saug' ich aus freier Welt;</l>
<l n='3' real="--+---+"> Wie ist Natur so hold und gut,</l>
<l n='4' real="--+---+"> Die mich am Busen haelt!</l>
<l n='5'> Die Welle wieget unsern Kahn</l>
<l n='6'> Im Rudertakt hinauf,</l>
<l n='7'> Und Berge, wolkig himmeln,</l>
<l n='8'> Begegnen unserm Lauf.</l>
</lgl>
```

On the other hand, the famous inserted alexandrine in Pope’s “Essay on Criticism”, might be encoded as follows:

```

<l n='356'> A needless alexandrine ends the song, </l>
<l n='357' met='-+|-+|-+|-+|-+|-+' real='++|-+|-+|-+|-+'>
  That, like a wounded snake, drags its slow length along.
</l>

```

Here the met attributes indicates that a different metrical convention (the alexandrine) is in force, while the real attribute indicates that there is a variation from that convention. As with many other aspects of metrical analysis, however, this is of necessity an entirely interpretive judgment.

#### 9.4.2 Segment-Level versus Line-level Tagging

The examples given so far have encoded information about the realization of metrical conventions at the level of the whole verse-line. This has obvious advantages of simplicity, but the disadvantage that any deviation from metrical convention is not marked at its precise point of occurrence in the text. Greater precision may be achieved, but only at the cost of marking deviant metrical units explicitly. This may be done with the <seg> element, giving the variant realization as the value of the real attribute on that element. Using this method, the example given immediately above might be encoded as follows:

```

<l n='356'> A need<seg type='foot' n='2' real='--'>less
  a</seg>alexandrine ends the song,</l>
<l n='357' met='-+|-+|-+|-+|-+|-+'>
  <seg n='1' real='++'> That, like </seg> a wounded snake,
  <seg n='4' real='++'> drags its </seg>
  <seg n='5' real='++'> slow length </seg>
  along.
</l>

```

The marking of the foot boundaries with the symbol | in the met attribute value of the <l> element allows the human reader, or a sufficiently intelligent software program, to isolate the correct portion of that attribute value as the default value for the same attribute on the <seg> elements for feet, namely --. It is of course up to the encoder to decide whether or not to include the n attribute of <seg> here, and whether or not also to tag the feet in the line in which there is no deviation from the metrical convention. The ability of software to infer which foot is being marked, if not all are tagged, will depend heavily on the language of the text and the knowledge of prosody built into the software; the fuller and more explicit the markup, the easier it will be for software to handle it. It may prove useful, however, to mark metrical deviations in the manner shown, even if the available software is not sufficiently intelligent to scan lines without aid from the markup. Human readers who are interested in prosody may well be able to exploit the markup in useful ways even with less sophisticated software.

There are circumstances where it may also be useful to use the met attribute of <seg>. If we wish to identify the exact location of the different types of foot in the first line of Virgil's *Aeneid*, the text could be encoded as follows (for simplicity's sake the caesura has been omitted):

```

<l><seg type='foot' met='+--'>Arma vi</seg>
  <seg met='+--'>rumque ca</seg>
  <seg met='++'>no Tro</seg>
  <seg met='++'>iae qui </seg>
  <seg met='+--'>primus ab</seg>
  <seg met='++'> oris</seg>
</l>

```

An appropriate value of the met attribute might also be supplied on the enclosing <div> element, to indicate that each foot may be made up of a dactyl or a spondee, so that the values given here for met at the level of the foot may be considered a series of local variations on this fundamental pattern; in cases like this, of course, the local variations may also be considered aspects of realization rather than of convention, in which case the real attribute may be used instead of met, if desired.

#### 9.4.3 Metrical Analysis of Stanzaic Verse

The method described above may be used to encode quite complex verse forms, for instance various kinds of fixed-form stanzas. Let us take one of Dante's canzoni, in which each stanza except the last has the same combination of eleven-syllable and seven-syllable lines, and the same rhyme scheme:

```

<div0 type='canzone'
  met='E/E/S/E/S/E/S/E/S/E/S/E/S/E/S/E/S/E'

```

```

    rhyme="abbcaccbdceeffghhhgg">
<lg1 n='1' type='stanza'>
<l n='1'>Doglia mi reca nello core ardire</l>
<!-- ... -->
</lg1></div0>

```

Here the `met` attribute specifies a metrical pattern for each of the twenty-one lines making up a stanza of the *canzone*. Each stanza inherits this definition from the parent `<div0>` element. The `rhyme` attribute specifies a rhyme scheme for each stanza, in the same way.

In the metrical notation used here, the letter E represents a line containing nine syllables which may or may not be metrically prominent, a tenth which is prominent and an optional non-prominent eleventh syllable. The letter “S” is used to represent a line containing five syllables which may or may not be metrically prominent, a sixth which is prominent and an optional non-prominent seventh syllable. A suitable definition for this notation might be given by a `<metDecl>` element like the following:

```

<metDecl type='met' pattern='((E|S)/)+'>
  <symbol value='E' terminal='N'>xxxxxxxx+o</symbol>
  <symbol value='S' terminal='N'>xxxxx+o</symbol>
  <symbol value='x'>metrically prominent or non-prominent</symbol>
  <symbol value='+'>metrically prominent</symbol>
  <symbol value='o'>optional non prominent</symbol>
  <symbol value='/'>line division</symbol>
</metDecl>

```

As noted above, the metrical pattern specified on the `<div0>` applies to each `<lg>` (stanza) element contained within the `<div0>`. In fact however, after seven stanzas of this type, there is a final stanza, known as a *commiato* or *envoi*, which follows a different metrical and rhyming scheme. The solution to this problem is simply to specify a new `met` attribute on the eighth stanza itself, which will override the default value inherited from parent `<div0>`, as follows:

```

<div0 met='....'>
  <!-- ... -->
  <lg1>
    <!-- This line group inherits its met value -->
    <!-- from the containing <div0> -->
    <l> ... </l>
  </lg1>
  <lg1 type='commiato'
    met='E/S/S/E/S/E/E/S/S/E/E'
    rhyme="abbcdeeedd">
    <l n='1'>Canzone, presso di qui &egrave; una donna</l>
    <!-- ... -->
  </lg1>
</div0>

```

Note that, in the same way as for the real attribute, over-riding of this kind does not affect subsequent elements at the same hierarchic level. Any `<lg1>` element following the *commiato* above would be assumed to use the same metrical and rhyming scheme as the one preceding the *commiato*. Moreover, although it is quite regular (in the sense that the last stanza of each *canzone* is a *commiato*), the over-riding must be specified for each case.

## 9.5 Rhyme

The `rhyme` attribute is used to specify the rhyme pattern of a verse form. Like the `met` attribute, it can be used with a user-specified notation documented by the `<metDecl>` element in the TEI header. Unlike `met`, however, the `rhyme` attribute has a default notation; if this default notation is used, no `<metDecl>` element need be given.

The default notation for rhyme offers the ability to record patterns of rhyming lines, using the traditional notation in which distinct letters stand for rhyming lines. For a work in rhyming couplets, like the Pope example above, the `rhyme` attribute simply specifies `aa`, indicating that pairs of adjacent lines rhyme with each other. For a slightly more complex scheme, applicable to groups of four lines, in which lines 1 and 3 rhyme, as do lines 2 and 4, this attribute would have the value `abab`. The traditional Spenserian stanza

has the pattern ababbcbbc, indicating that within each nine line stanza, lines 1 and 3 rhyme with each other, as do lines 2, 4, 5 and 7, and lines 6, 8 and 9.

Non-rhyming lines within such a group may be represented using a hyphen or an x, as in the following example:

```
<lg rhyme='AB-BBA'>
<l>The sunlight on the garden</l>
<l>Hardens and grows cold,</l>
<l>We cannot cage the minute</l>
<l>Within its nets of gold</l>
<l>When all is told</l>
<l>We cannot beg for pardon.</l>
</lg>
```

Note however that the default notation includes no specific way of recording ‘internal’ rhyme, such as that between the end of the first line and the start of the second in this particular poem. For this, a special user-defined notation would need to be declared using the `<metDec1>` element in the header. Alternatively, rhyme, like alliteration or assonance, may be considered as a special form of ‘correspondence’, and hence encoded using the mechanisms defined for that purpose in section 14.4 *Correspondence and Alignment*.

To use the correspondence mechanisms to represent the complex rhyming pattern of the above example, we first need to delimit and identify each rhyming sequence within the text: the `<seg>` element may be used for this purpose, as follows:

```
<lg rhyme='AB-BBA'>
<l>The sunlight on the <seg id='A1'>garden</seg></l>
<l><seg id='A2'>Harden</seg>s and grows <seg id='B1'>cold,</seg></l>
<l>We cannot cage the <seg id='C1'>minute</seg></l>
<l>Wi<seg id='C2'>thin it</seg>s nets of <seg id='B2'>gold</seg></l>
<l>When all is <seg id='B3'>told</seg></l>
<l>We cannot beg for <seg id='A3'>pardon</seg>.</l>
</lg>
```

Now that each rhyming word, or part-word, has been tagged and allocated an arbitrary identifier, the general purpose `<link>` element may be used to indicate which of the `<seg>` elements share the same rhyme, as follows:

```
<linkGrp type='rhyme'>
<link targets='A1 A2 A3' />
<link targets='B1 B2 B3' />
<link targets='C1 C2' />
</linkGrp>
```

For further discussion of the `<link>` and `<linkGrp>` element, see section 14.4 *Correspondence and Alignment*.

## 9.6 Encoding Procedures For Other Verse Features

A number of procedures that may be of particular concern to encoders of verse texts are dealt with elsewhere in these guidelines. Some aspects of layout and physical appearance, especially important in the case of free verse, are dealt with in chapter 18 *Transcription of Primary Sources*. Some initial recommendations for the encoding of phonetic or prosodic transcripts, which may be helpful in the analysis of sound structures in poetry, are to be found in chapter 11 *Transcriptions of Speech*; it may also be found convenient to use standard entity names (those proposed for the International Phonetic Alphabet suggest themselves) to mark positions of suprasegmentals such as primary and secondary stress, or other aspects of accentual structure.

As already indicated, chapter 14 *Linking, Segmentation, and Alignment* contains much which will be found useful for the aligning of multiple levels of commentary and structure within verse analysis. Encoders of verse (as of other types of literary text) will frequently wish to attach identifying labels to portions of text that are not part of a system of hierarchical divisions, may overlap with one another, and/or may be discontinuous; for instance passages associated with particular characters, themes, images,

allusions, topoi, styles, or modes of narration. Much of the computerized analysis of verse seems likely to require dividing texts up into blocks in this way. The `<span>` element discussed in 15.3 *Spans and Interpretations* provides the means for doing this. Finally, the procedures for the tagging of feature structures, described in chapter 16 *Feature Structures*, provide a powerful means of encoding a wide variety of aspects of verse literature, including not only the metrical structures discussed above, but also such stylistic and rhetorical features as metaphor.

For other features it must for the time being be left to encoders to devise their own terminology. Elements such as `<metaphor tenor="..." vehicle="..."> ... </metaphor>` might well suggest themselves; but given the problems of definition involved, and the great richness of modern metaphor theory, it is clear that any such format, if pre-defined by these Guidelines, would have seemed objectionable to some and excessively restrictive to many. Leaving the choice of tagging terminology to individual encoders carries with it one vital corollary, however: the encoder must be utterly explicit, in the TEI header, about the methods of tagging used and the criteria and definitions on which they rest. Where no formal elements are currently proposed, such information may readily be given as simple prose description within the `<encodingDesc>` element defined in section 5.3 *The Encoding Description*.

