MODULE 4: POETRY

Edward Vanhoutte

Ron Van den Branden

Melissa Terras
# TABLE OF CONTENTS

3. Rhyme.............................................................................................................................................................................. 1
  3.1 Rhyming Words....................................................................................................................................................... 1
  3.2 Rhyme Patterns....................................................................................................................................................... 2
  3.3 Rhyming Words and Patterns............................................................................................................................... 3
3. Rhyme

So far, all of the elements and attributes for structuring verse texts were members of the “common” elements and attributes defined in the core TEI module. Yet, TEI provides a specific verse module as well, which defines a number of elements and attributes specific for the encoding of verse texts. Some of these will be discussed in the following sections. In order to use them, a TEI schema must include all (or just the required) components of that verse module; see Module 8: Customising TEI, ODD, Roma for a tutorial on how to customise TEI.

3.1 Rhyming Words

The rhyming words of a line of verse can be encoded using the appropriate <rhyme> element:

```xml
<l xmlns="http://www.tei-c.org/ns/1.0" type="poem">
  <lg type="stanza">
    <l><rhyme>Poppadom</rhyme></l>
    <l><rhyme>Oatmeal</rhyme></l>
    <l>Bubble <rhyme>gum</rhyme></l>
    <l>Cut of <rhyme>veal</rhyme></l>
  </lg>
  <lg type="stanza">
    <l>Mince for <rhyme>pie</rhyme></l>
    <l>Frozen <rhyme>peas</rhyme></l>
    <l>Video for <rhyme>Guy</rhyme></l>
    <l>Selection of <rhyme>teas</rhyme></l>
  </lg>
  <lg type="stanza">
    <l>Paper towels/garbage <rhyme>bags</rhyme></l>
    <l>Pasta sauce and <rhyme>Parmesan</rhyme></l>
    <l>Pumpkin seed and olive <rhyme>oil</rhyme></l>
  </lg>
  <lg type="stanza">
    <l>Cheesy crisps and favourite <rhyme>mags</rhyme></l>
    <l>Kidney beans (1 large <rhyme>can</rhyme>)</l>
</lg>
```
Module 4: Poetry

Example 15. Encoding rhyme words with `rhyme`).

`rhyme` can appear anywhere in the line. This way, not only end-of-line rhymes can be tagged but also internal rhyme, even inside prose(-like) paragraphs like in the following fragment:

```xml
<p xmlns="http://www.tei-c.org/ns/1.0">This <rhyme>course</rhyme> on verse <rhyme>is</rhyme> <rhyme>terse</rhyme> and provides a <rhyme>fine</rhyme> design <rhyme>for the study of poetry like <rhyme>yours</rhyme> and <rhyme>mine</rhyme></p>
```

Example 16. `rhyme` can be used inside prose paragraphs, too.

3.2 Rhyme Patterns

Rhyme patterns can be documented with a `@rhyme` attribute which has a default notation in which distinct letters stand for rhyming lines. This attribute can be added to `<lg>` and/or to `<l>`, and also to any `<div>` element that is used for the encoding of poetry. Of course, the

The rhyme scheme in the shopping list poem is `ababcdcdefefg`. This can be documented inside the `@rhyme` attribute of `<lg type="poem">`. The rhyme scheme of the separate stanzas can be encoded inside the `@rhyme` attribute of `<lg type="stanza">`, and even the rhyme scheme of the separate lines can technically be encoded inside the `@rhyme` attribute of the `<l>` element. The complexity of the use of all these options depends on the encoder. A maximally complex encoding could be the following:

```xml
<lg xmlns="http://www.tei-c.org/ns/1.0" type="poem" rhyme="ababcdcdefefg">
  <lg type="stanza">
    <l rhyme="a">
      <rhyme>Poppadom</rhyme>
    </l>
    <l rhyme="b">
      <rhyme>Oatmeal</rhyme>
    </l>
    <l rhyme="c">
      <rhyme>Tomatoes</rhyme>
    </l>
    <l rhyme="d">
      <rhyme>Carrots</rhyme>
    </l>
    <l rhyme="e">
      <rhyme>Coriander</rhyme>
    </l>
    <l rhyme="f">
      <rhyme>Asparagus</rhyme>
    </l>
    <l rhyme="g">
      <rhyme>Bread</rhyme>
    </l>
  </lg>
</lg>
```
Example 17. Encoding the rhyme scheme of different structural units, with @rhyme.

3.3 Rhyming Words and Patterns

The correspondence between the rhyming pattern documented in the @rhyme attribute and the rhyming words encoded with the <rhyme> element can be specified in a @label attribute on the <rhyme> element. The value of this attribute is usually one of the letters of the rhyme pattern. Applied to the shopping list poem, this results in the following encoding:
Example 18. Identifying a `<rhyme>` in a rhyme scheme with `@label`.

All `<rhyme>` elements with the same value for their `@label` attribute are assumed to rhyme with each other within a given scope. That scope is defined by the nearest ancestor element for which the `@rhyme` attribute has been supplied.

In the following encoding of the same poem, the scope is defined by the nearest ancestor element with a `@rhyme` attribute, i.e., the `<lg type="stanza">` element. This means that the rhyming words labelled a, b, or c are only assumed to rhyme inside that stanza and not across stanzas:
Example 19. The nearest ancestor element with a `@rhyme` attribute determines the scope for `@label`.

**SUMMARY**

The occurrence of rhyming words and rhyming patterns and their correspondence can be encoded by a combination of tags and attribute values. Depending on the encoder’s preferences, they can be applied to different structural levels of the text.