

## Contents

<b>1</b>	<b>Linking and Analysis</b>	<b>1</b>
1.1	Objectives . . . . .	1
1.2	Getting starting . . . . .	1
1.3	Importing a taxonomy . . . . .	1
1.4	Linking to your taxonomy categories . . . . .	3
1.5	Using ana for more classification . . . . .	3
1.6	Investigating linguistic markup . . . . .	4
<b>2</b>	<b>Application to your own project</b>	<b>4</b>
2.1	Creating your own taxonomy . . . . .	4

## 1 Linking and Analysis

### 1.1 Objectives

The aim of this exercise is to give you some familiarity in the use of linking and analysis. You will

- work with an existing file
- learn about `<taxonomy>`
- get practice adding `@ana` attributes
- Think about taxonomies for your own project.

### 1.2 Getting starting

To start with load up the file `collection.xml`. This is a very brief collection with some text samples in it. It contains a haiku, a shakespearean sonnet, a speech from Hamlet, and a transcription of an interview.

Investigate the file:

- Note its minimal header with a blank `<taxonomy>`
- Look at the divisions. These could, of course, been done as separate `<text>` elements inside a `<group>`, or indeed `<TEI>` elements inside a `<teiCorpus>`, but that seemed like overkill for this brief collection divisions.
- Make sure you understand the markup contained in the file.

### 1.3 Importing a `<taxonomy>`

A `<taxonomy>` element gives you a structured way to provide a hierarchical set of categories and their classifications. In general they look something like:

```
<taxonomy>
  <category xml:id="overallCategory">
    <catDesc>Overall Description</catDesc>
    <category xml:id="mainCategory1">
      <catDesc>Main Category 1</catDesc>
      <category xml:id="subCategory1">
        <catDesc>Sub Category 1</catDesc>
        <category xml:id="subSubCategory1">
          <catDesc>Sub Sub Category 1</catDesc>
        </category>
      </category>
    </category>
    <category xml:id="mainCategory2">
      <catDesc>Main Category 2</catDesc>
    </category>
  </category>
</taxonomy>
```

It would be boring to have to type in a taxonomy ourselves, so one has been made for you. Load up the file `taxonomy.xml` in oXygen. Highlight the whole of the `<taxonomy>` and copy and paste it into `collection.xml` over existing empty `<taxonomy>` element. (There are other ways to do this, such as inserting the file at the right place using the Document, File, Insert File menu.)

Explore the `<taxonomy>` you've added. There are three sections. The first is general literary categorisation:

```
<!-- General literary categorisation --><category xml:id="literature">
  <catDesc>Literature</catDesc>
  <category xml:id="poetry">
    <catDesc>Poetry</catDesc>
    <category xml:id="sonnet">
      <catDesc>Sonnet</catDesc>
      <category xml:id="shakesSonnet">
        <catDesc>Shakespearean Sonnet</catDesc>
      </category>
    </category>
  </category>
<!-- ... -->
</category>
</category>
```

The second is for metrical analysis:

```
<!-- Meter --><category xml:id="meter">
  <catDesc>Metrical Categories</catDesc>
  <category xml:id="feet">
    <catDesc>Metrical Feet</catDesc>
    <category xml:id="iambic">
      <catDesc>Iambic</catDesc>
    </category>
    <category xml:id="trochaic">
      <catDesc>trochaic</catDesc>
    </category>
  </category>
<!-- ... -->
</category>
```

The third is for linguistic analysis

```

<!-- linguistic analysis --><category xml:id="pos">
  <catDesc>part of speech analysis</catDesc>
  <category xml:id="adje">
    <catDesc>adjectives</catDesc>
    <category xml:id="AJ0">
      <catDesc>adjective (unmarked) (e.g. GOOD, OLD)</catDesc>
    </category>
    <category xml:id="AJC">
      <catDesc>comparative adjective (e.g. BETTER, OLDER)</catDesc>
    </category>
  </category>
<!-- ... -->
</category>
</category>

```

You can create `<taxonomy>` elements for absolutely anything, using any categories and heiarchies that you want. This makes them extremely flexible.

## 1.4 Linking to your `<taxonomy>` categories

One of the ways to link to categories in your taxonomy is to use the global `@ana` attribute. This can appear on any element and takes 1 or more whitespace-separated URIs. In practice, if the `<taxonomy>` is in your own file this means just prefixing the `<category>` element's `@xml:id` attribute with a `#` to indicate a fragment URI.

If you look in the literature `<category>` of the `<taxonmy>` you will notice a `<category>` for poetry and inside that one for haiku. Move down to the haiku in the body of the document and mark the division as a haiku by adding an `@ana` attribute to the division as a whole. Don't forget the `#` on the id of the category. This should end up looking like:

```

<div ana="#haiku">
  <head>Haiku</head>
  <lg>
    <l>The silent old pond </l>
    <l>a mirror of ancient calm, </l>
    <l>a frog-leaps-in splash. </l>
  </lg>
</div>

```

## 1.5 Using `@ana` for more classification

The `@ana` attribute can be used for multiple classifications, and these can refer to widely different categories or indeed taxonomies. Mark up the sonnet as a shakespearean sonnet (`#shakesSonnet`), being in iambic pentameter (`#iambic #pentameter`). Also mark each of that stanzas (`#stanza`) except the last one which is a couplet (`#couplet`). The result should be something like:

```

<div ana="#shakesSonnet #iambic #pentameter">
  <head>Sonnet 18</head>
  <lg ana="#stanza">
    <l>Shall I compare thee to a summer's day?</l>
    <l>Thou art more lovely and more temperate.</l>
    <l>Rough winds do shake the darling buds of May,</l>
    <l>And summer's lease hath all too short a date.</l>
  </lg>
  <lg ana="#stanza">

```

```
<l>Sometime too hot the eye of heaven shines,</l>
<l>And often is his gold complexion dimmed;</l>
<l>And every fair from fair sometime declines,</l>
<l>By chance, or nature's changing course, untrimmed;</l>
</lg>
<lg ana="#stanza">
  <l>But thy eternal summer shall not fade,</l>
  <l>Nor lose possession of that fair thou ow'st,</l>
  <l>Nor shall death brag thou wand'rest in his shade,</l>
  <l>When in eternal lines to Time thou grow'st.</l>
</lg>
<lg ana="#couplet">
  <l>So long as men can breathe, or eyes can see,</l>
  <l>So long lives this, and this gives life to thee.</l>
</lg>
</div>
```

Similarly, you could mark up the following speech from Hamlet as being drama (#drama):

```
<div ana="#drama">
  <head>To be, or not to be</head>
  <stage rend="italic center"
    type="entrance">Enter Hamlet.</stage>
  <sp>
<!-- ... -->
  </sp>
</div>
```

While we're at it. Mark up the interview that follows as both spoken (#spoken) and an interview (#interview). This is of course redundant. By pointing to #interview we are already including all of its ancestors, which includes its parent #spoken. This should look like:

```
<div ana="#spoken interview">
  <head>Spoken text</head>
<!-- ... -->
</div>
```

## 1.6 Investigating linguistic markup

The final division in the collection is a spoken text interview between Stuart Lee and Ian Hislop. This has been marked up with the parts of speech for each word. This is done with the @ana attribute which points back to our <taxonomy> in the header. Follow several of the references back up to the linguistic <category> and make sure you understand the parts of speech of the first utterance <u> element. Consider what uses this sort of mark up might have. What new questions might it enable someone to ask?

Clearly this markup would be painful to add manually. It would be much better to generate it and then correct any errors, and indeed that is precisely what was done, using freely available part of speech tagging software online. This acts as a good reminder that often some degree of markup can be provided semi-automatically and it is worth investigating how this might be done (or who might do it) for your own resources.

## 2 Application to your own project

Your own project may wish to categorise and analyse certain aspects of the texts it encodes.

### 2.1 Creating your own taxonomy

Think about the documents of your own project:

- What overall categories of analysis might you want to record?
- How can these be broken into subcategories?
- How deep should this nesting of categories go?
- Should each document have its own taxonomy or should you include (for example using XInclude) an overall one?

Make a list of the categories and subcategories you would use in your own project.