

TEI BY EXAMPLE



MODULE 8: CUSTOMISING TEI, ODD, ROMA

Ron Van den Branden

Edward Vanhoutte

Melissa Terras

Centre for Scholarly Editing and Document Studies (CTB) , Royal
Academy of Dutch Language and Literature, Belgium, Gent, 9 July 2010

Last updated September 2020

Licensed under a Creative Commons Attribution ShareAlike 3.0 License

TABLE OF CONTENTS

1. jTEI: a Customisation for Journal Articles.....	1
2. Abolishing Numbered Text Divisions.....	6

1. jTEI: a Customisation for Journal Articles

This examples provides some snippets of the jTEI customisation ODD file, which defines a minimalist TEI scheme for encoding journal articles. This customisation is being used for encoding the source files of the articles of the [Journal of the Text Encoding Initiative](#). This “clean” customisation defines a trimmed-down model of TEI texts, making it easier to enforce a consistent encoding style. This way, authors and editors are offered guidance in the composition and editing of an article. Inside the `<schemaSpec>` section of this "tei_j tei" customisation, you'll see that only a limited number of elements is included from the **core**, **corpus**, **figures**, **header**, **namesdates**, **tagdocs**, **tei**, **textstructure**, and **transcr** TEI modules.

In this example, the definition of the `<figure>` element is made more stringent, in order to enforce a very specific structure on its contents. This is done inside an `<elementSpec>` element with a value of "figure" for the `@ident` attribute; the `@mode` attribute indicates that the existing element element specification should be changed. The contents for the `<figure>` element are being redefined in a `<content>` element, which is specifying a very strict *sequence* of either a single `<graphic>`, `<egXML>`, or `<eg>` element, which must be followed by one or more `<head>` elements. Notice how the `<elementRef>` elements are being used to refer to these different TEI elements, and the `<sequence>` and `<alternate>` elements are being used to specify how they are combined. With the `@minOccurs` and `@maxOccurs` attributes on the `<elementRef>` reference to the `<head>` element, it is stipulated that at least 1 `<head>` element must be present, without an upper bound. The `<attList>` section is used to remove the attributes `@place` and `@title` from `<figure>`.

Likewise, the specification of the `<titleStmt>` element is re(de)efined, in that its `<content>` is reduced to a sequence of one or more `<title>` elements, followed by one or more `<author>` elements. Notice that the order is important here: there's no `<alternate>` wrapper here, that would allow alternation of these elements. Notice, too, how this `<elementSpec>` element contains additional modifications: a `<constraintSpec>` element is used to define additional *Schematron* checks that can express context-dependent validation rules. In this case, a rule is defined which ensures that at least one `<title>` element must be present in `<titleStmt>`, with a value "main" for its `@type` attribute. For more information on the definition of such Schematron constraints in ODD: see section [22.5.2 Additional Constraints](#) of the TEI Guidelines.

```

<TEI xmlns="http://www.tei-c.org/ns/1.0" xmlns:xsl="http://www.w3.org/1999/XSL/
Transform" xmlns:sqf="http://www.schematron-quickfix.com/validator/process"
xmlns:sch="http://purl.oclc.org/dsdl/schematron" xmlns:tei="http://www.tei-c.org/
ns/1.0" xml:lang="en">
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title>jTEI input customization</title>
        <author xml:id="RvdB">
          <name>Ron Van den Branden</name>
          <email>ron.vandenbranden@kantl.be</email>
        </author>
        <author xml:id="MDH">
          <name>Martin Holmes</name>
          <email>mholmes@uvic.ca</email>
        </author>
      </titleStmt>
      <editionStmt>
        <edition>Originally generated on <date>2014-01-30T13:29:15Z</date> using
          oddbyexample.xsl. Progressively modified after that. </edition>
      </editionStmt>
      <publicationStmt>
        <p/>
      </publicationStmt>
      <sourceDesc>
        <p>Originally generated by oddbyexample.xsl, based on analyzing 6 files
          from file:/F:/ctb/jTEI/jTEIsource/branches/working docs/tei2openedition/
          oddbyexample. Manually modified thereafter. </p>
        </sourceDesc>
      </fileDesc>
      <!-- ... -->
    </teiHeader>
    <text>
      <body>
        <!-- prose documentation -->
      </body>
      <back>
        <!-- ... -->
      </back>
    </text>
  </TEI>

```

```

<div>
  <schemaSpec ident="tei_jtei" start="TEI">
    <moduleRef key="tei"/>
    <moduleRef key="core" include="abbr author bibl biblScope cit date desc editor
      email emph foreign gap graphic head hi item label lb list listBibl mentioned
      name note num p pubPlace publisher q quote ptr ref resp respStmt series so
      Called term title"/>
    <moduleRef key="corpus" include="particDesc"/>
    <moduleRef key="figures" include="cell figure row table"/>
    <moduleRef key="header" include="appInfo application availability catRef
      change classCode edition encodingDesc fileDesc idno keywords langUsage
      language licence listChange profileDesc projectDesc publicationStmt rendition
      revisionDesc seriesStmt sourceDesc tagsDecl teiHeader textClass titleStmt"/>
    <moduleRef key="namesdates" include="affiliation forename listPerson person
      placeName orgName roleName surname"/>
    <moduleRef key="tagdocs" include="att code eg egXML gi ident tag val"/>
    <moduleRef key="textstructure" include="TEI back body div front text"/>
    <moduleRef key="transcr" include="supplied"/>
    <!-- ... -->
    <elementSpec ident="titleStmt" module="header" mode="change">
      <gloss versionDate="2005-01-14" xml:lang="en">title statement</gloss>
      <desc versionDate="2014-03-10" xml:lang="en">groups information about the
        title of a work and those responsible for its content. In jTEI, this must
        include a title and an <gi>author</gi> element for each author of the
        paper, which in turn must include <gi>name</gi>, <gi>affiliation</gi>
        and <gi>email</gi>.</desc>
      <content>
        <!-- [RvdB] sequence wrapper seems necessary to keep ODD2DTD happy -->
        <sequence>
          <elementRef key="title" minOccurs="1" maxOccurs="unbounded"/>
          <elementRef key="author" minOccurs="1" maxOccurs="unbounded"/>
        </sequence>
      </content>
    <constraintSpec ident="jtei.sch-title" scheme="schematron">
      <constraint>
        <sch:rule context="tei:titleStmt">
          <sch:assert test="tei:title[@type = 'main']" sqf:fix="type.add"> A title
            of type "main" is required in <sch:name/>. </sch:assert>

```

```

    <sqf:fix id="type.add">
      <sqf:description>
        <sqf:title>Add a @type='main' attribute to the first
          title.</sqf:title>
      </sqf:description>
      <sqf:add match="tei:title[not(@type='main')][1]" node-
        type="attribute" target="type">main</sqf:add>
    </sqf:fix>
  </sch:rule>
</constraint>
</constraintSpec>
</elementSpec>
<!-- ... -->
<elementSpec xmlns:rng="http://relaxng.org/ns/
structure/1.0" ident="figure" mode="change">
  <gloss xml:lang="en" versionDate="2015-03-03">figure</gloss>
  <desc xml:lang="en" versionDate="2015-03-03">groups elements representing
    or containing graphic information such as a graphic illustration, or a
    block of example code. Figure must contain either <gi>graphic</gi>, <gi>eg
    XML</gi> (for example XML code) or <gi>eg</gi> (for non-XML code), and a
    mandatory <gi>head</gi> element containing the caption for the figure.
    Inline code examples may be provided through <gi>egXML</gi> and <gi>eg</gi>
    without the <gi>figure</gi> wrapper. </desc>
  <content>
    <!-- [RvdB] sequence wrapper seems necessary to keep ODD2DTD happy -->
    <sequence>
      <alternate>
        <elementRef key="graphic"/>
        <elementRef key="egXML"/>
        <elementRef key="eg"/>
      </alternate>
      <elementRef key="head" minOccurs="1" maxOccurs="unbounded"/>
    </sequence>
  </content>
  <attList>
    <attDef ident="place" mode="delete"/>
    <attDef ident="type" mode="delete"/>
  </attList>

```



```

<exemplum xml:lang="en">
  <egXML xmlns="http://www.tei-c.org/ns/Examples">
    <figure>
      <graphic url="http://www.example.org/fig1
        .png" width="100px" height="750px"/>
      <head type="legend">The View from the Bridge</head>
      <head type="license">Used with permission</head>
    </figure>
  </egXML>
</exemplum>
<exemplum xml:lang="en">
  <egXML xmlns="http://www.tei-c.org/ns/Examples">
    <figure>
      <eg>if ((err = SSLHashSHA1.update(&hashCtx, &serverRandom)) != 0) goto
        fail; if ((err = SSLHashSHA1.update(&hashCtx, &signedParams)) != 0)
        goto fail; goto fail; if ((err = SSLHashSHA1.final(&hashCtx, &hash
        Out)) != 0) goto fail; </eg>
      <head type="legend">An embarrassing error for Apple</head>
    </figure>
  </egXML>
</exemplum>
<exemplum xml:lang="en">
  <egXML xmlns="http://www.tei-c.org/ns/Examples">
    <figure> <egXML xmlns="http://www.tei-c.org/ns/Examples"> <list rend=
      "bulleted"> <item>Life</item> <item>The Universe</item> <item>Everything<
      /item> </list> </egXML> <head type="legend">A book title rendered as a
      list</head></figure>
    </egXML>
  </exemplum>
</elementSpec>
<!-- ... -->
</schemaSpec>
</div>
</back>
</text>
</TEI>

```

Example 1. A snippet of the jTEI customisation ODD file (Van den Branden and Holmes, 2014).

TEI XML source available from https://tei-c.org/release/xml/tei/custom/odd/tei_jtei.odd.

2. Abolishing Numbered Text Divisions

Following ODD file defines a minimal TEI schema (containing the **core**, **tei**, **header**, and **textstructure** modules). Besides that, it does away with numbered text divisions (<div1>, ..., <div7>) by specifying the value "delete" for the @mode attribute on their respective <elementSpec> declarations.

Notice that strictly speaking, only deletion of <div1> is necessary, as all further numbered subdivisions should nest according to the TEI scheme. By definition, if the topmost level is deleted, all further subdivisions are practically excluded from the document model. Although they are still declared, they can't be used in documents according to this schema. Additionally, notice how this list of separate <elementSpec> instructions could be expressed more concisely as:

```
<moduleRef xmlns="http://www.tei-c.org/ns/1.0" key="textstructure" except="div1 div2
div3 div4 div5 div6 div7"/>
```

```
<TEI xmlns="http://www.tei-c.org/ns/1.0" xml:lang="en">
  <teiHeader>
    <fileDesc>
      <titleStmt>
        <title>numbered divisions</title>
        <author>The TBE crew</author>
      </titleStmt>
      <publicationStmt>
        <p>for use by whoever wants it</p>
      </publicationStmt>
      <sourceDesc>
        <p>no source, born digital</p>
      </sourceDesc>
    </fileDesc>
  </teiHeader>
  <text>
    <front>
```

```
<divGen type="toc"/>
</front>
<body>
  <p>This customisation only selects the minimal TEI modules, and deletes all
    numbered subdivisions.</p>
  <schemaSpec ident="numDiv" xml:lang="en">
    <moduleRef key="core"/>
    <moduleRef key="tei"/>
    <moduleRef key="header"/>
    <moduleRef key="textstructure"/>
    <elementSpec module="textstructure" ident="div1" mode="delete"/>
    <elementSpec module="textstructure" ident="div2" mode="delete"/>
    <elementSpec module="textstructure" ident="div3" mode="delete"/>
    <elementSpec module="textstructure" ident="div4" mode="delete"/>
    <elementSpec module="textstructure" ident="div5" mode="delete"/>
    <elementSpec module="textstructure" ident="div6" mode="delete"/>
    <elementSpec module="textstructure" ident="div7" mode="delete"/>
  </schemaSpec>
</body>
</text>
</TEI>
```

Example 2. TBE-crafted example encoding, no material source.

BIBLIOGRAPHY

Van den Branden, Ron and Martin Holmes. 2014. "Journal of the Text-Encoding Initiative Article Schema. Schema and guidelines for encoding an article for the journal." <https://tei-c.org/guidelines/customization/jtei/>.