

Schema Planning

Instructions

This worksheet is intended to guide you through some of the analytic process needed for an effective schema design. It assumes that you've already performed an initial document analysis, for instance using the worksheet at

http://www.wwp.neu.edu/outreach/seminars/_current/handouts/document_analysis_worksheet_1.pdf

It may also be helpful to complete the project analysis process described here:

for instance using the worksheet at

http://www.wwp.neu.edu/outreach/seminars/_current/handouts/document_analysis_worksheet_2.pdf

The questions below are aimed at helping you plan your TEI schema customization and work through the initial stages of its development.

Schema Planning

Test document

In order to test out your encoding ideas (and to test whether your evolving schema is working properly) it's helpful to create a test document. This could be a sample document from your project. However, an even better strategy is to create a composite document purely for testing purposes, that contains representative samples of all the features you've identified as important in your document analysis. This test document will have several advantages over a real document:

- You can guarantee that it will contain all of the features you're interested in
- You can remove unnecessary length and duplicative features, making the document easier to work with and faster to validate
- You can use it to test alternative encoding scenarios without any lingering unease about brutalizing your real documents

To create the test document, use the TEI-All template (either downloaded from Roma or created directly within Oxygen). For each feature in your inventory, transcribe a representative sample of that feature into the test document. At the outset, encode these as plainly and simply as you can. As you work your way through the schema development process, you will progressively modify your test document (and your schema) to refine the encoding and its constraints. When the process is completed, the test document will ideally represent a sample encoding that can be used for documentation, training, and reference.

Features of the transcription

[These questions are aimed at helping you identify the features that will be included in your schema, and how they should be represented.]

For each of the features you identified in the initial document analysis, if you haven't already identified the specific elements needed to produce an appropriate encoding, do so now. Update your test document as you go so that it matches your inventory of features and elements.

For each of the elements you've identified, consider what specific TEI attributes will be required, and whether you will need to specify a controlled vocabulary or datatype, or alter an existing TEI controlled vocabulary or datatype for that attribute. Update your test document as you go so that each element carries the appropriate attributes with accurate test values.

In any cases where you identified a specific configuration of elements, consult the TEI Guidelines (or try it out in your test document) to see whether this configuration can be encoded using the TEI in unmodified form, or whether a customization is needed to allow specific elements in specific places.

For any features that require the creation of new elements, add examples of these elements to your test document. They'll be invalid until you customize your schema, but the invalidity will serve as a reminder of work to be done.

Features not present

[To trim down your schema, you should eliminate elements and attributes that won't be used. These questions are aimed at helping you eliminate unnecessary components from your schema. You'll need to consult the TEI Guidelines for a full list of the elements in each module, and for a full list of attributes available for each element. You can also ascertain what attributes are present on specific elements just by consulting your test document: try inserting an attribute and see what's on the drop-down list.]

What TEI elements will never be needed in your encoding? Are they in TEI modules (i.e. chapters of the Guidelines) that contain elements you do plan to use? Or can you eliminate the entire module that contains them?

In cases where you need some (but not all) of the elements in a given module, consider whether you may want your schema to include new TEI elements if these are introduced later on. (For instance, if the TEI introduces a new name component to go inside <persName>, do you want that included in future versions of your schema?) If you want your schema to respond to future changes in the TEI, then you should use the @except attribute on <moduleRef> to exclude unwanted elements from the module. If you want your schema to remain stable through changes to the TEI, then you should use the @include attribute to specify which elements you want to include from each module.

What TEI attributes will never be needed in your encoding? Are there entire attribute classes that can be removed from the schema?

Schema modification

[The following is a suggested process for modifying your TEI schema. You may find other approaches work better for you and if so you should use them.]

1. Identify the modules you need to include, and add a <moduleRef> for each one.
2. For each module, include or exclude elements as necessary following your analysis above.

3. Generate a schema and use it to validate your test document. If your document is valid, you'll know that you haven't inadvertently removed any essential elements. If you find you need to restore elements to the schema to validate the test document, do so now.
4. If your customization plan includes any new elements, create those now and revalidate your document.
5. If your customization plan requires that any elements be permitted in places where they are not allowed in TEI-All, you'll need to modify the schema to permit this. It will probably involve adding these elements to one or more additional model classes. Make these changes gradually, one at a time, and revalidate your document after each change.
6. If your customization plan involves any heightened constraints on elements (for instance, to prevent elements from appearing in a certain context or to trim down content models), you'll need to modify the schema to permit this. It will probably involve removing elements from model classes. As above, make these changes very gradually and carefully, and revalidate your document after each change.
7. Once your preliminary element selection and overall document structuring is complete, you can turn your attention to attributes. For any attribute for which you identified a controlled value list, add that list to your customization. If you haven't already done so, add test cases for each attribute to your test document and revalidate your test document.
8. Remove any attribute classes that you believe to be unnecessary. Revalidate your test document. At this stage, you will probably find that you've eliminated one or two attributes you are actually using and you'll have to restore them.
9. Next, check individual elements to see what individual attributes need to be removed from the attribute classes you've retained. (For instance, you may need to keep `att.type` in your schema if you're using `@type`, but you might want to remove `@subtype` from the class if you're not planning to use it.) Remove these attributes and revalidate your test document.
10. Try encoding an actual sample document using your schema. This process will probably reveal places where your customization has omitted a feature you need; as you find these, make further schema modifications and revalidate both the sample document and your test document. In addition, as you encode the sample, notice what elements are allowed at each place in your encoding (and think about the decisions that other encoders would be making in your position). If there are irrelevant elements available that someone might use by mistake, consider removing these from the schema or from the content model. When you've finished encoding the sample and it's valid, it will become another test document to use in verifying future schema changes.
11. Repeat step 10 with further sample documents, making additional schema modifications as needed and revalidating all of your accumulated test documents after each change. The testing process is finished when you can successfully encode any new document using the schema, and when anyone else using the schema (plus whatever documentation you provide) can accurately and consistently encode documents for your project.