

Stand-Off conf call Feb. 1, 2018

Hangout

https://hangouts.google.com/hangouts/_/brown.edu/teicouncil

Present

- Peter Stadler
- Syd Bauman
- Elli Mylonas
- Hugh Cayless
- Martina Scholger
- Raffaele Viglianti
- Laurent Romary

Apologies: James Cummings

Documents

<https://github.com/laurentromary/stdfSpec/tree/AnnArbor>

Wake up, standOff! [Slides:] <https://hal.inria.fr/hal-01374102>

Introduction by Laurent

- LR compiled Stand-Off use cases at <https://github.com/laurentromary/stdfSpec/blob/AnnArbor/Scenarios/StandOffScenarios.xml>
- only introduces 2 new elements <standoff>, <listAnnotation>
- Standoff elements should be aligned with W3C Web Annotation Model <https://www.w3.org/TR/annotation-model/>
- separate TEI document for standoff with its own TEI header - this can be paired with the original document or allow the original to continue in its pristine form.
- stand-off got linked to <ldb> at some point but it's not intended to be

Discussion

Is there room in <standoff> for other TEI stand-off like tags, for ex. linkGrp - interp - LR asserts yes, there is.

- **LR to expand ODD and documentation to clarify how this will work**

Q: Why include the teiHeader within standOff?

A: standOff should work stand-alone as well as within a TEI document
council should concentrate on how to handle OAbody

A few alternatives were discussed, including:

- Use TEI/teiHeader when a new TEI file is created just for annotations (ie annotating other TEI files) OR modify existing TEI/teiHeader if the annotated file is writable.
 - Counterargument: this makes it harder to add/remove annotations and their metadata easily (which could be solved by wrapping the annotated document and the annotating document in a teiCorpus, but LR says this makes the process too complicated and may hinder adoption).
- <TEI> should be able to nest, possibly making <standOff> unnecessary (though council seems to agree that having <standOff> at the same level as other resources such as <text> and <facsimile> makes sense).
 - A few agreed that nesting <TEI> may be useful nonetheless, suggesting this topic is discussed at some other time and independently of the standOff element proposal.

Q (RV): If other stand-off elements (think interpGrp, linkGrp) will be part of annotationBlock's content model, we're saying that these forms of stand-off are all annotations. Are we cool with that?

OABody and OATarget should be fleshed out and how <interp> etc fit into <annotationBlock>

- Afterthought from RV: while it's true that these elements are reproducing OA's model, I don't think TEI elements ever include another standard's name within a their name/@ident. Can we think of something generic, while still crediting OA in <desc> and/or <remarks>, as well as the Guidelines?
- [LR]: agree. J'ai forcé le trait, as we say in my language to clearly indicate the mapping.
- [HC]: Beyond that objection, I worry that what we have isn't quite modeling the semantics of OA's data model (see <https://www.w3.org/TR/2017/REC-annotation-model-20170223/#bodies-and-targets>). I think that's something we could certainly do in TEI, but I'm not sure the proposal is doing it. In particular, I'd have thought that with regard to , if you wanted to think of it in an OA fashion, @from and @to would mark the oa:target, and the content of the span would be the oa:body.
- [LR] The whole idea of the model is to have a triple target-annotation-body, where annotation is a centre hinge allowing an NxM flexible relationship between target and body. Putting the body as content for (with the semantics of) to be the target, you completely lose this combinatorial possibility (e.g. multiple discontinuous spans being annotated together). <interp> is the trick to implement this. You could chose other elements, but you need the mechanism. And as said, <interp> is handy because its semantics is in the same realm and was quite lasy/dormant so far.
- [HC] I get all that, and I agree with the idea. What I'm concerned with is coming up with a model where the semantics of the TEI and OA are congruent. Again, I'm concerned about as the oa:target because its semantics seem to me to

model a (very basic) whole OA Annotation. Wouldn't it be better to use elements whose semantics concern just pointing in oa:target? <ptr> comes to mind...

[April 11, MT] <https://github.com/laurentromary/stdfSpec/issues/20>

Pre-meeting Apr 29th notes [MT]

Zoom meeting location: <https://dariah.zoom.us/j/959180068>

Use-cases & WIP examples

1. standoff version of people/places/org identification
 - a. <persName ref="#JH #CH"><w xml:id="w15">brothers</w></persName>
 - b. https://github.com/tuurma/stdfSpec/blob/AnnArbor/Samples/NER-Jon_and_Charles.xml
 - c. multiple fragments of the transcription reference the same entity; example should illustrate how to avoid redundant entries and ideally how to link with existing TEI mechanisms of providing canonical identification for people, also for cases where it is not a part of the document being annotated
 - d. text fragment may reference multiple different entities; example should illustrate how to encode such cases in unambiguous way
 - e. each single annotation, regardless if created and maintained by human or software agent should have possibility to associate relevant metadata: agent responsible, time of creation/update etc; example should illustrate these
 - f. annotations may but don't necessarily have to reside within the document being annotated; example (or its variant) should illustrate how to create standoff annotations for external documents
2. [HC] identification of a string as (e.g.) a personal name
 - a. <placeName ref="https://pleiades.stoa.org/places/993">Gallia</placeName>
 - b. <https://gist.github.com/hcayless/93318af9e36e4a3eccf51f66200ef8dd>
 - c. [MT] not sure what's the difference between this example and the 1st one. Maybe if places were just marked, without identification (TEI equiv <placeName>Gallia</placeName>) the difference would be clearer but still not sure it was HC intention [HC] The difference is that your example is associating a person with a string in the text body, while mine is asserting that the string in the text can be read as a <placeName>. I think both are legitimate styles of annotation, and that both might be things people would want to do.
3. events
 - a. <https://github.com/laurentromary/stdfSpec/blob/AnnArbor/Scenarios/StandOffScenarios.xml#L222-L258>
 - b. <https://hedgehog-web.herokuapp.com/json2xml/index.html>
 - c. link to the full example doesn't work

- d. relationships between multiple annotation targets for the same body should be explicit: does the example suggest that both spans describe the same event (but are separate descriptions) or that it is a single description, just not continuous within the original text
- 4. morphological annotation
 - a. <https://github.com/laurentromary/stdfSpec/blob/AnnArbor/Scenarios/StandOffScenarios.xml#L268-L371>
- 5. critical apparatus
 - a.

General questions/thoughts

[HC] standoff/listApp [HC] I don't want debate over this to consume too much time—it's just an idea, but we should think about what can go in <standoff>. I'm currently thinking (almost?) all list* elements.

[MT] Ideally examples should present different perspectives on the same text (as far as possible). Eg entity recognition and MAF annotation could be illustrated with the same base text, with a bit of case choosing the initial text general commentaries and events could be covered as well

[MT] I would like the content model of annotationBlock to be clearly sub-divided into parts representing annotation-body-target

[HC] annotationBlock currently has a very specific use and content model. Do we feel comfortable extending that massively? What would the content model of annotationBlock look like? AnnotationBlock already has @start and @end attributes. Presumably those should be forbidden if it's being used in web annotation mode?

Apr 29th Meeting summary

keep standoff as a name, no ldb !important

exact content model for standoff tbd as long as it contains listAnnotation

new annotation element for the 'annotation hinge'

span for targets (or surface, or...)

listAnnotation may contain annotationBlocks or new annotation elements

annotationBlock may contain new annotation element or current content model