Why TEI stand-off annotation doesn't quite work

Piotr Bański

Institute of English Studies,
University of Warsaw
bansp at o2.pl

Balisage: the Markup Conference, 5 August 2010

Why TEI stand-off annotation doesn't quite work

and why you might want to use it nevertheless

Piotr Bański

Institute of English Studies,
University of Warsaw
bansp at o2.pl

Balisage: the Markup Conference, 5 August 2010

Presentation vs. paper

Reviewers: be more general! Me: OK! :-)

Things I will not do in this presentation (with apologies to my wonderful reviewers):

- present an introduction to the TEI (I wouldn't dare, not here...),
- compare TEI stand-off with other approaches (I might mention a few though),
- speculate about using RDF for stand-off annotation – it gives me the power of graphs and the pain of graphs; in this presentation, I'd rather focus on trees;
- generalize on the failure of XPointer... beyond a few general statements,
- review other, non-linguistic applications in which XPointer or XInclude has [not] been adopted and the reasons why (it's an open-ended challenge that requires expertise in the possibly numerous particular fields – I'm not quite up to it... yet).

But I want to mention the above, because they are all valid questions (and I'm proud that my paper generated them!)

Presentation – sketch

Instead, I will skip even more stuff from the paper, and talk about:

- a linguist's view of overlapping hierarchies,
- annotation layers and how they are supposed to interact in the TEI...
- … and about how they do not interact...
- and about why it may be sensible and enlightened to hug an OWL.

The story will touch upon some of the seemingly disparate issues taken up in the paper, and end in a few shallow, commonplace, Sunday-fair-type generalizations. And a plug.

Presentation – sketch

Instead, I will skip even more stuff from the paper, and talk about:

- a linguist's view of overlapping hierarchies,
- annotation layers and how they are supposed to interact in the TEI...
- … and about how they do not interact...
- and about why it may be sensible and enlightened to hug an OWL.

The story will touch upon some of the seemingly disparate issues taken up in the paper, and end in a few shallow, commonplace, Sunday-fair-type generalizations. And a plug.

But first...


Warning:

Warning: there be dragons

Warning: there be dragons

(all the sleeping dragons of markup overlap)

Warning: there be dragons

(all the sleeping dragons of markup overlap)

(oh... let them lie...)

Limits of OHCO (overlap)

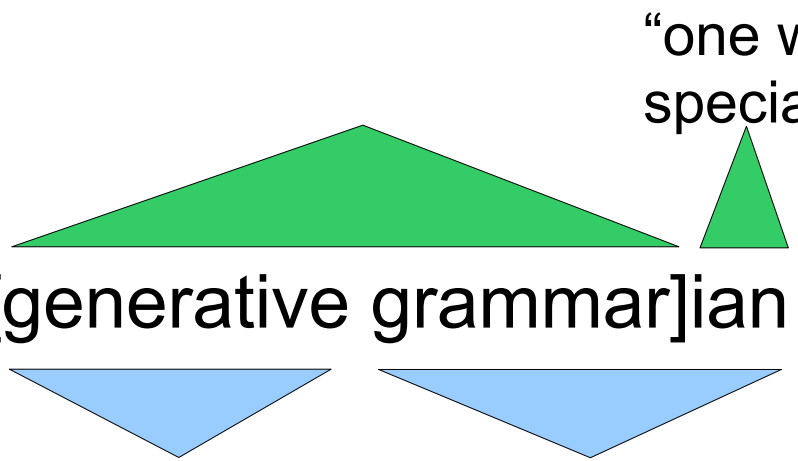
OHCO thesis: text is Ordered Hierarchy of Content Objects

[see e.g. S. DeRose, D. Durand, E. Mylonas, A. Renear (1990). “What is text, really?”]

However: verse vs. sentence, metrical vs. syntactic, line vs. speaker, etc., these hierarchies don't always nest – they often overlap;

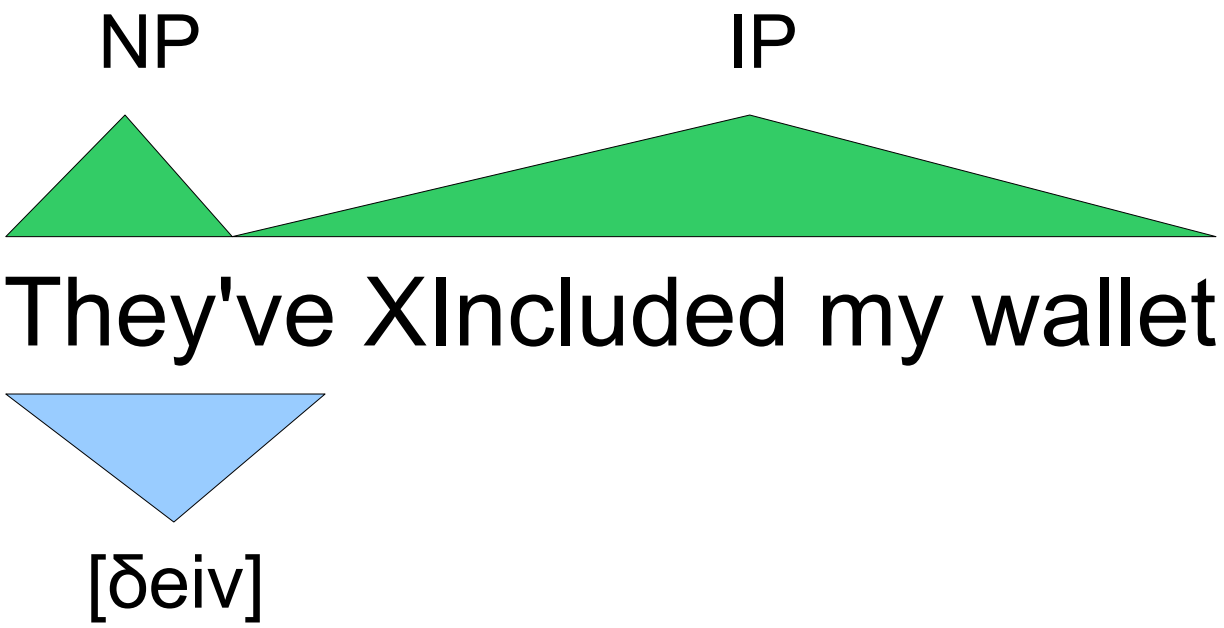
stand-off markup is one way to handle overlap.

And of course, linguistics offers more examples:



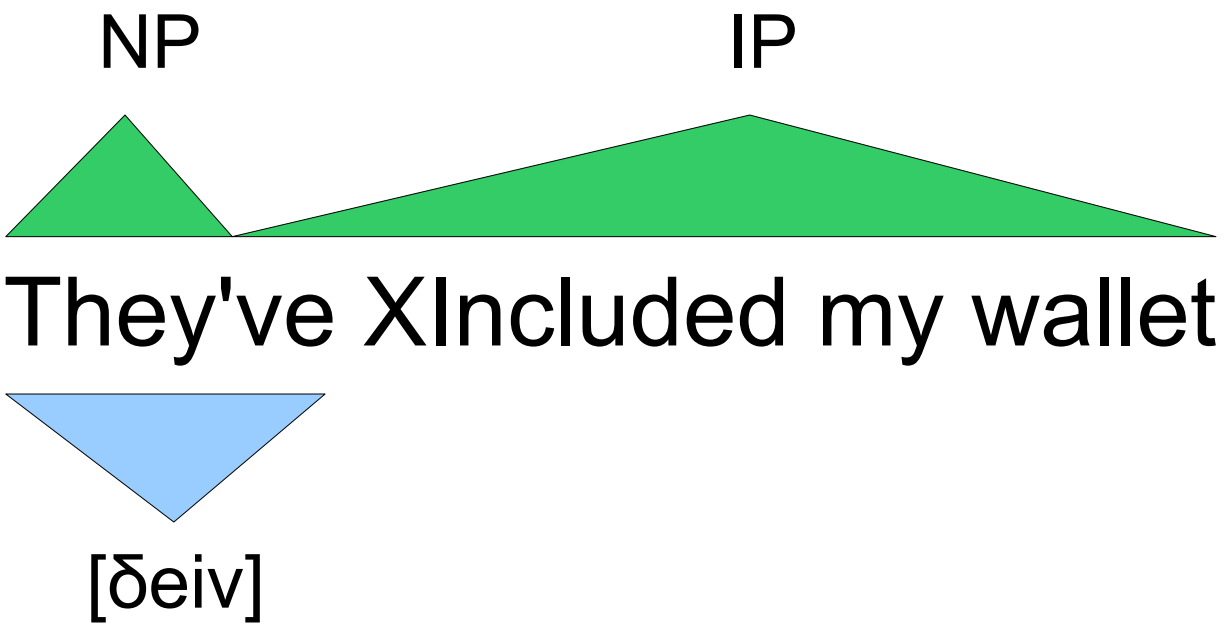
semantic vs. lexical units

Limits of OHCO (overlap)



syntactic phrasing (Noun Phrase and I(nflection) Phrase) vs. phonological phrasing

Limits of OHCO (overlap)



syntactic phrasing (Noun Phrase and I(nflection) Phrase) vs. phonological phrasing

BUT

~~OHCO failure~~ vs. modularity

It's not so much the failure of OHCO that governs the content of annotation layers in a corpus:

constraint A: keep conflicting hierarchies separate

constraint B: preserve the integrity of modules

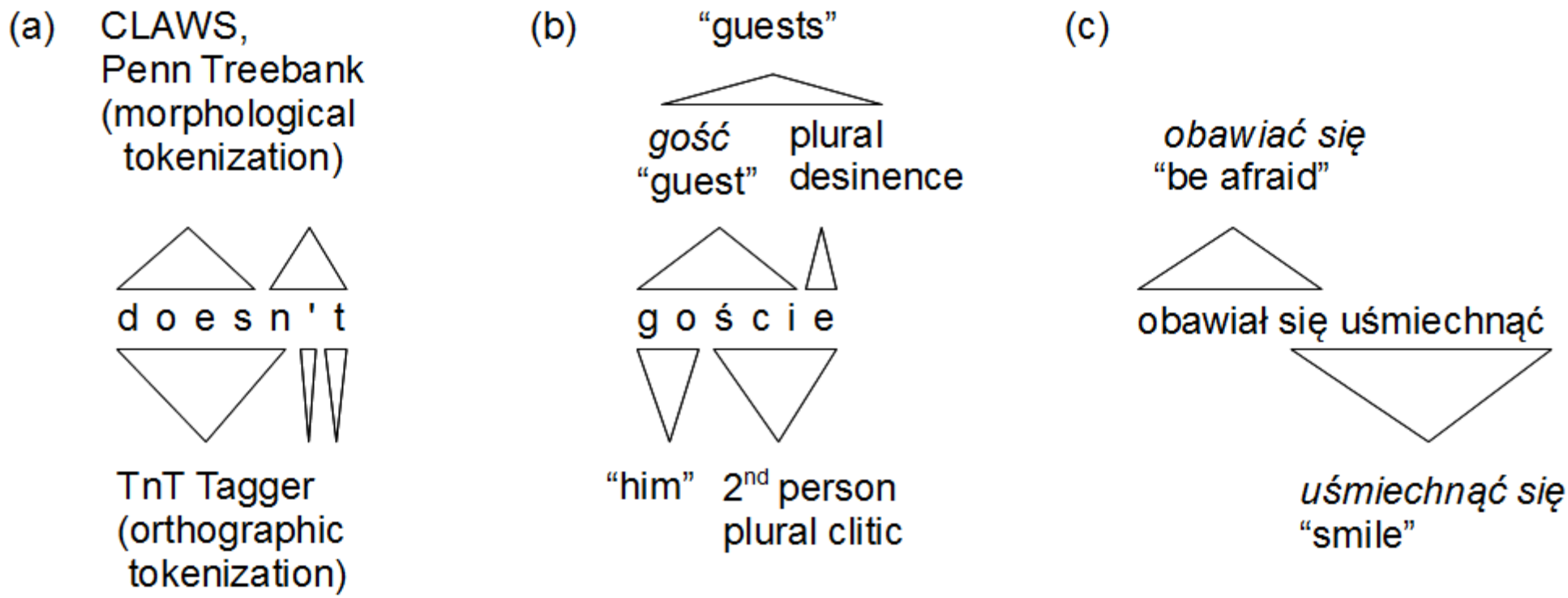
constraint B >> constraint A

("B is ranked higher than A", "B weighs more than A"

therefore

in case they conflict, preserve B at the cost of violating A)

Constraint conflict illustrated

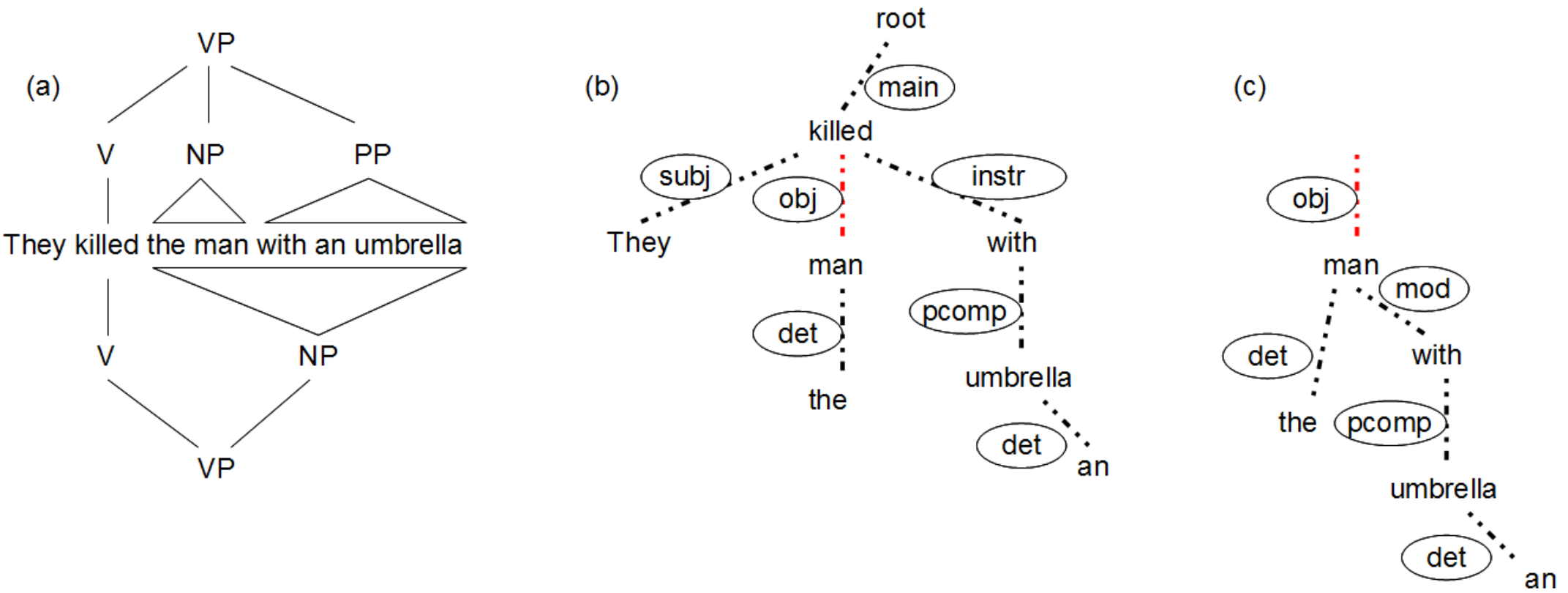


a: lexical segmentation b: lexical segmentation c: syntactic segmentation

Keep (a) separated, for practical reasons (see also next slide);

do not separate (b) or (c) = live with the failure of the OHCO thesis.

Different schools of thought (and formalisms)



The upper tree of (a) encodes the same reading as (b);
the lower tree of (a) encodes the same reading as (c).

We want to keep both trees of (a) together, and (b) and (c) in their own layer.

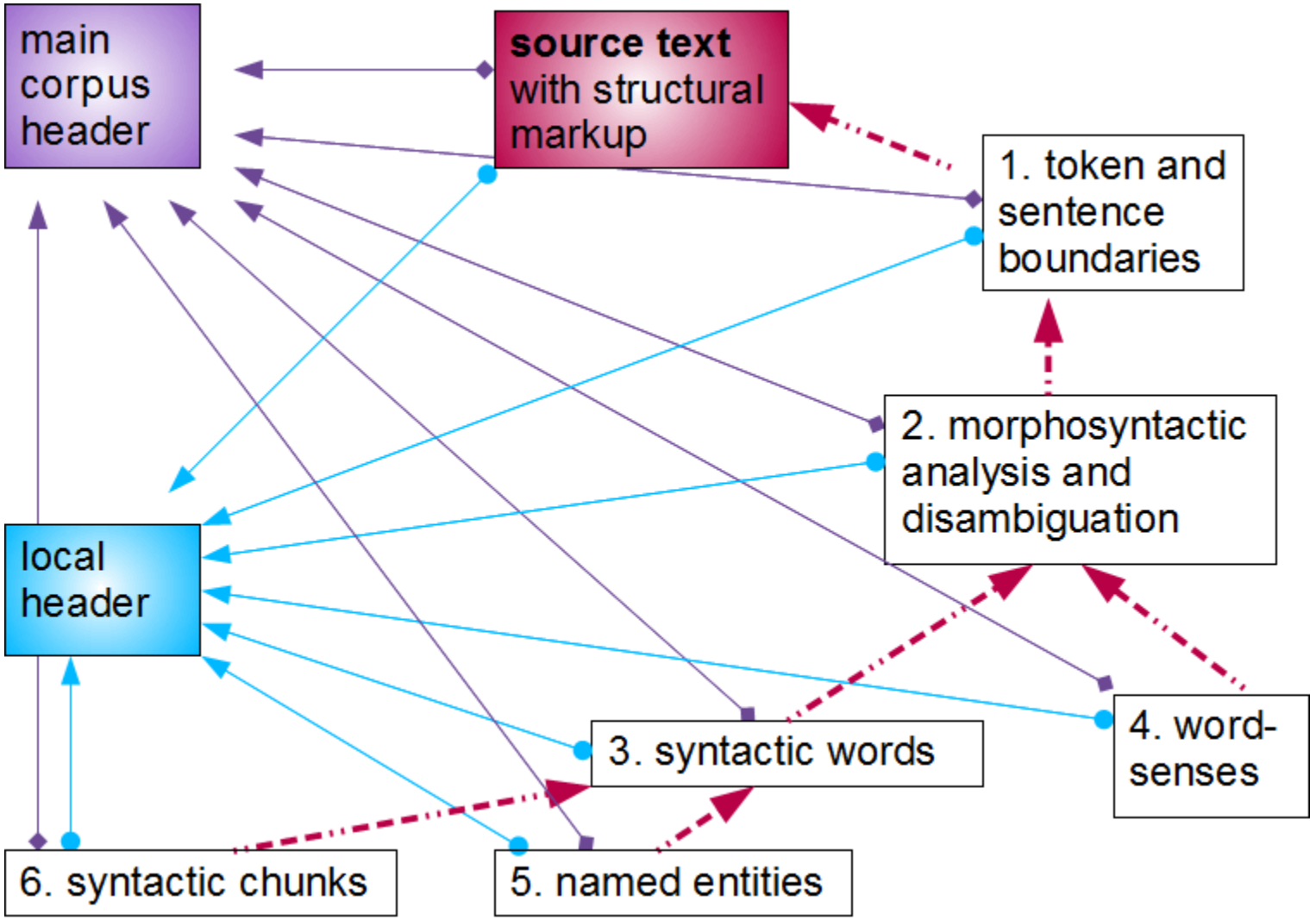
Summary at this point

When identifying annotation layers (= particular views of the data), maintain the integrity of modules:

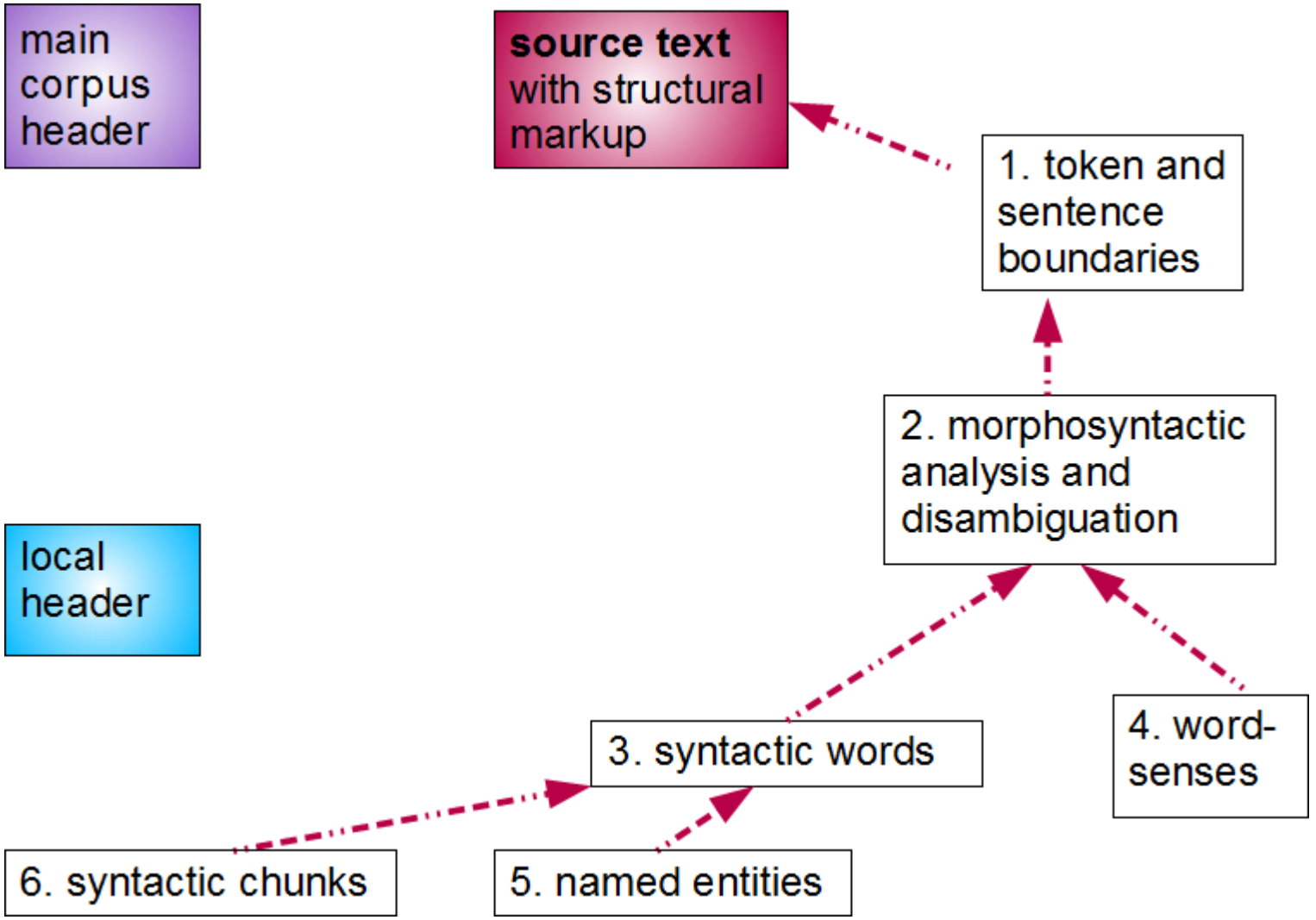
- violate OHCO but represent two “readings” within a single view? *keep them together anyway*
- belong to different views (including views created by different tools, different theoretical approaches, etc.)? *keep them separate*

Let's have a look at example layers now:

A picture. With colours!



National Corpus of Polish: annotation layers



How TEI stand-off works

morphosyntax.xml:

```
<seg>
   <seg><xi:include href="segmentation.xml" xpointer="seg with ID(xxx)"/></seg>
   <fs>(feature structure with all possible interpretations and disambiguation)</fs>
</seg>
```

parse morphosyntax.xml --xinclude

parse segmentation.xml

How TEI stand-off works

segmentation.xml:

```
<seg xml:id="xxx">
   <seg><xi:include href="base_text.xml"
            xpointer="5-char range beginning from the Nth char in that div/p"/></seg>
</seg>
```

parse morphosyntax.xml --xinclude

parse segmentation.xml

parse base_text.xml

How TEI stand-off works

base_text.xml:

```
<div>
    <p xml:id="NoamCh.">Colorless green ideas sleep furiously.</p>
</div>
```

parse morphosyntax.xml --xinclude

parse segmentation.xml

parse base_text.xml

(create the infoset)

How TEI stand-off works

segmentation.xml:

```
<seg xml:id="xxx">
   <seg><xi:include href="base_text.xml"
            xpointer="5-char range beginning from the Nth char in that div/p"/></seg>
</seg>
```

parse morphosyntax.xml --xinclude

parse segmentation.xml

parse base_text.xml

(pull in words from base text,
create the infoset)

(create the infoset)

How TEI stand-off works

segmentation.xml:

```
<seg xml:id="xxx">
   <seg>sleep</seg>
</seg>
```

parse morphosyntax.xml --xinclude

parse segmentation.xml

parse base_text.xml

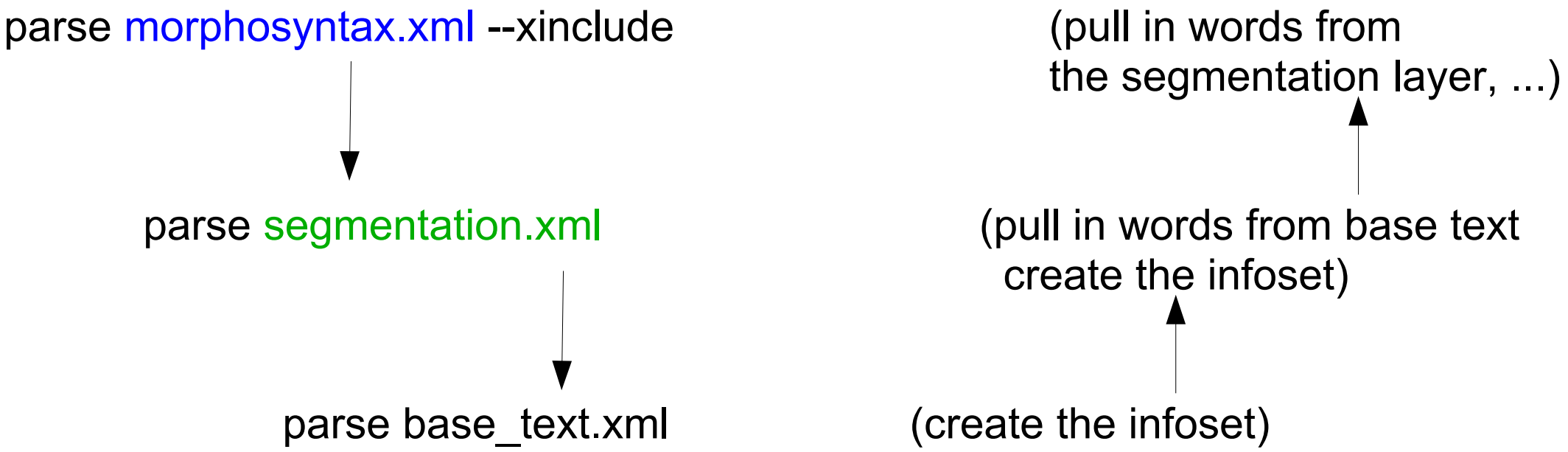
(pull in words from base text
create the infoset)

(create the infoset)

How TEI stand-off works

morphosyntax.xml:

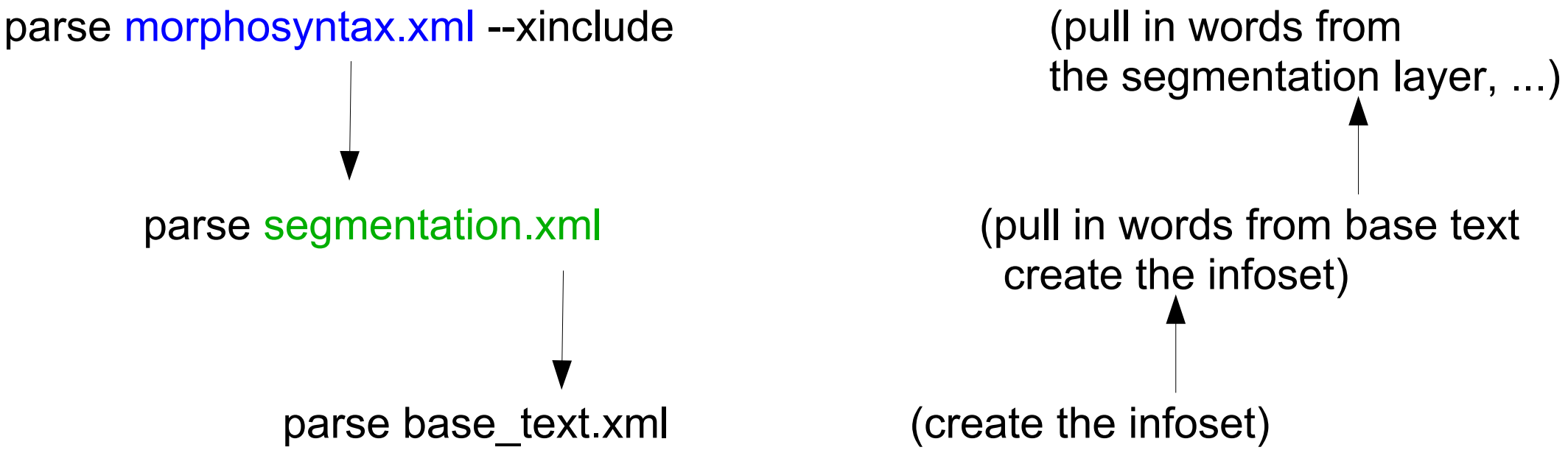
```
<seg>
   <seg><xi:include href="segmentation.xml" xpointer="seg with ID(xxx)"/></seg>
   <fs>(feature structure with all possible interpretations and disambiguation)</fs>
</seg>
```



How TEI stand-off works

morphosyntax.xml:

```
<seg>
   <seg>sleep</seg>
   <fs>(feature structure with all possible interpretations and disambiguation)</fs>
</seg>
```



How TEI stand-off works

morphosyntax.xml:

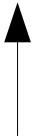
```
<seg>
   <seg>sleep</seg>
   <fs>(feature structure with all possible interpretations and disambiguation)</fs>
</seg>
```

parse morphosyntax.xml --xinclude

parse segmentation.xml

parse base_text.xml

(pull in words from
the segmentation layer, ...)



(pull in words from base text
create the infoset)

(create the infoset)

See? It's beautiful... And all of this for free!

(Why might you want to use TEI stand-off?)

Because it's beatiful... and comes for free, given generic XML tools.

How TEI stand-off doesn't work

Well, it fails wherever it depends on TEI-defined XPointer schemas.

What is wrong with the TEI-defined schemas?

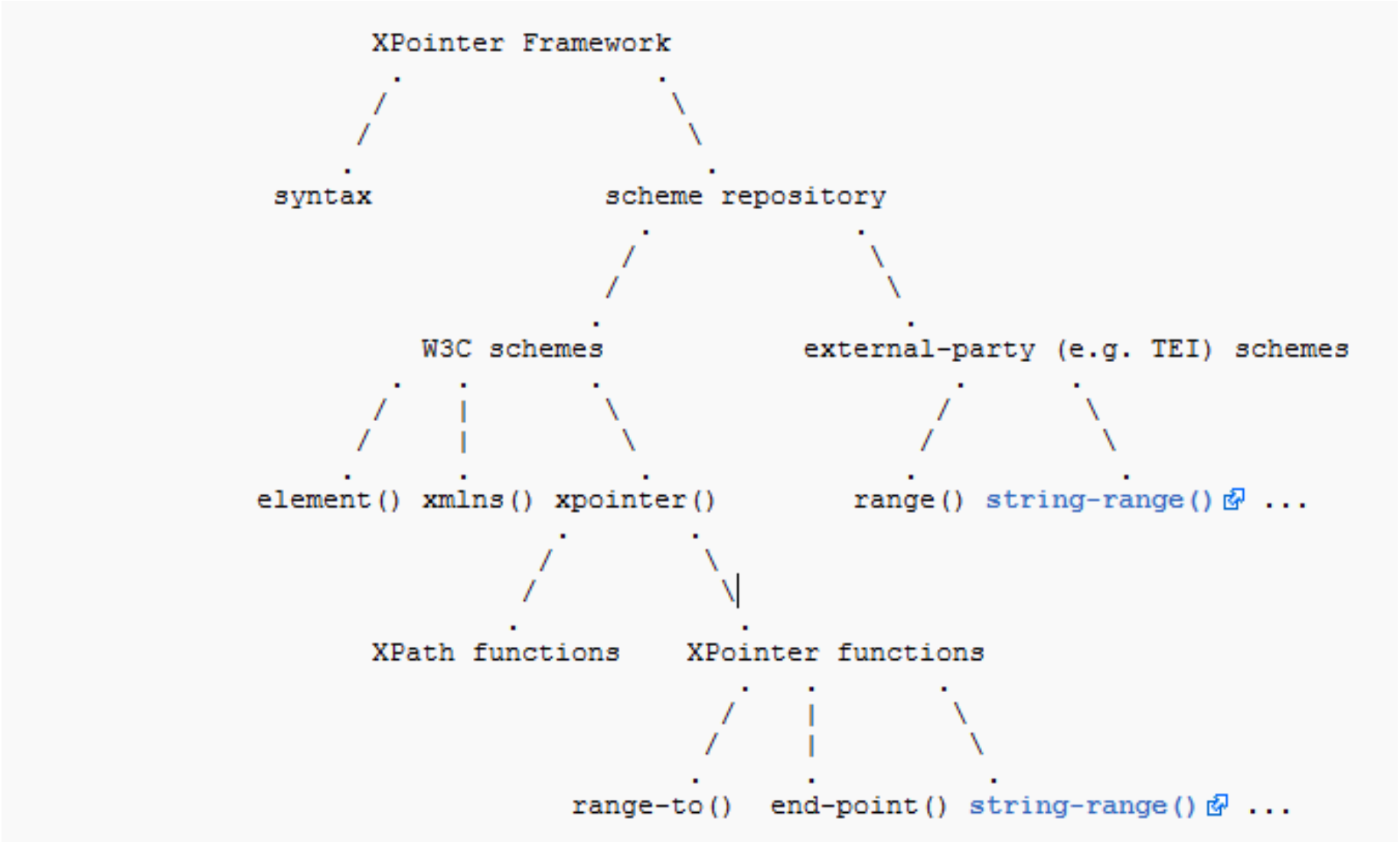
Nothing. There are simply no tools to make them work.

Any replacements? Oh yes, W3C XPointer xpointer() Working Draft, the dead one.

What to do next then?

- create a replacement (come to the talk by Hugh and Adam), or
- “it's not *dead* – it's just *dormant*... waky waky...”

Waking up a dormant draft...



Waking up a dormant draft

Good thing: xmlint almost handles XInclude with the W3C xpointer() scheme

Bad thing: the implementation is buggy

Good thing: XPointer Framework (the Recommendation) allows for cascading XPointers:

"TEI_scheme(...) xpointer(...) element(...)"

So you can keep both the TEI and the W3C pointers together, and if the TEI pointer fails, the W3C pointer will rescue you. Hehe. When it's fixed.

Why fix it? Because a crowd of linguists may then start to use it, and you want this crowd. Some of us are clever (don't look at me), and we have lots of data, lots of use cases, and will be a wonderful addition to your user base, oh XML Technologists and Standards Bodies... You can only gain.

"Lots of data?" Yeah. Our corpora now can be gathered straight from the Web. We just need a nice, fast and easy way to annotate them. For now, it isn't XML...

Back to the TEI for a moment

The TEI romanced with linguistics ever since the beginning:

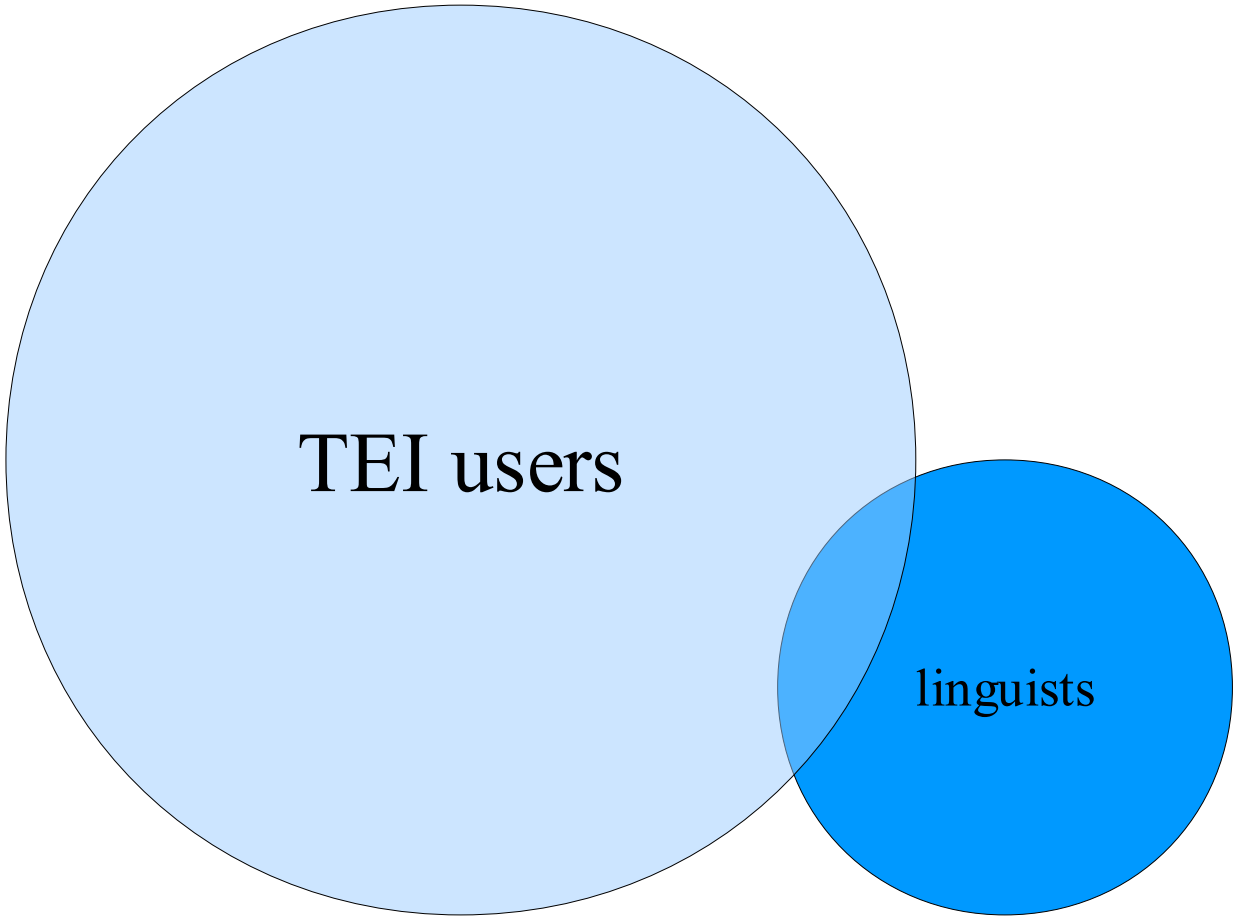
- the British National Corpus
- the TEI/ISO Feature Structure Recommendation
- the unfinished chapter on corpus annotation
- the finished chapter on multiple hierarchies
- now the ISO LAF – TEI interaction
- etc.

Still, maybe after the success of the (X)CES, there was less pressure on the TEI itself to elaborate on the good things for language-resource annotation. It's time to change it:

- a few low-level solutions in the paper,
- a high level incentive: new SIG in the TEI (it's happening).

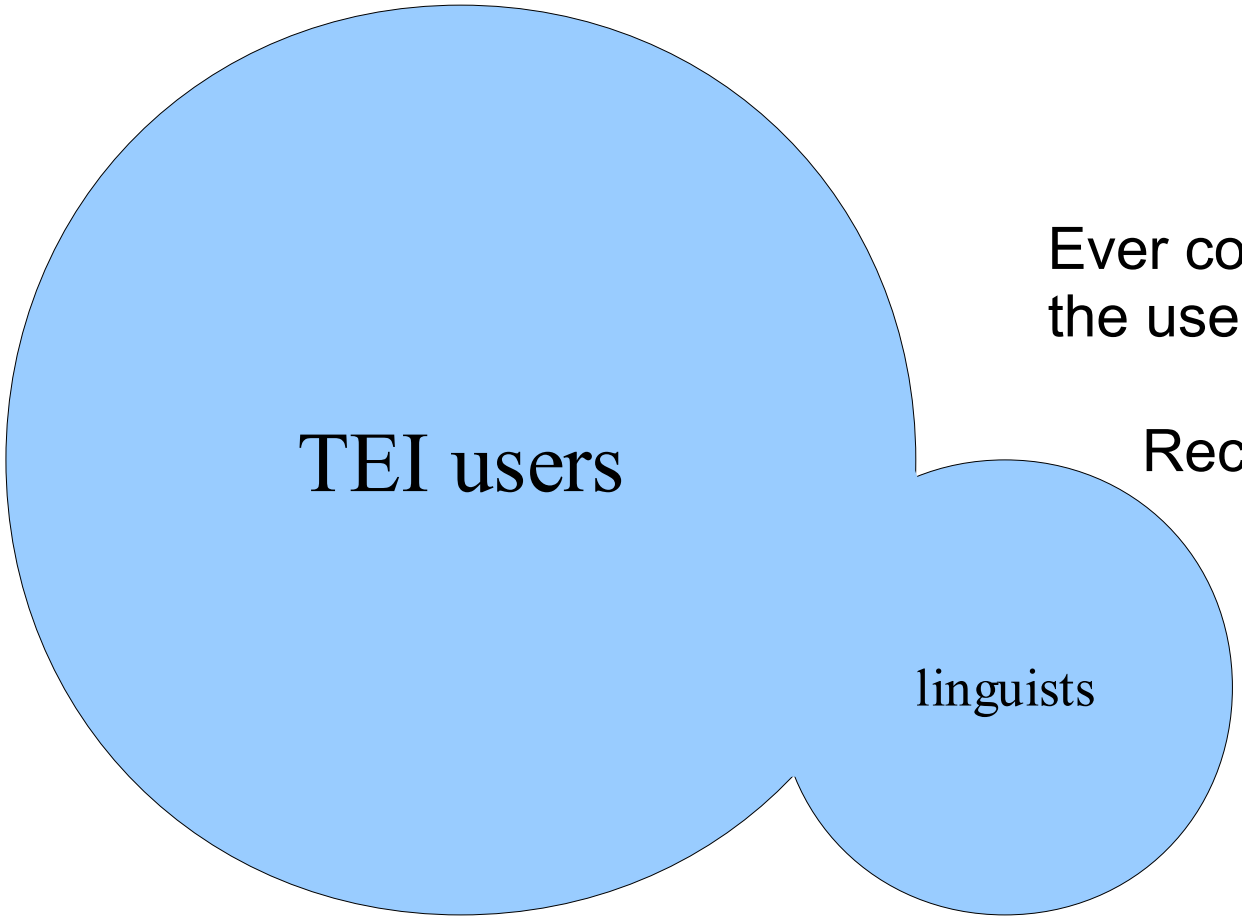
Serving them right...

TEI: solutions that work 90% of the time for 90% of the users...



Enlightened altruism

TEI: solutions that work 90% of the time for 90% of the users...



Ever considered enlarging
the user base, rapidly?

Recalculate the solutions...

Conclusions

1. There are a few things that are sub-optimal about the beautiful TEI stand-off recommendations
(one of them being that they just don't work in their canonical form)

BUT all is not lost

- be here when Hugh Cayless presents,
- do think of getting your student to fix xpointer.c in xmllint for us linguists, pleeease.

2. OHCO and its limits certainly played and play a role in the development of stand-off (and related) approaches, but there is another constraint which is more important:
modularity of description: let the given annotation layer be informationally uniform and distinct from others

3. **Enlightened altruism**: be open, try hard to accommodate promising communities in yours – and in time, they may make you richer.

Shameless (well, that's just me) plug:

TEI Members' Meeting and Conference
(Zadar, Croatia, November 8–14)

http://www.tei-c.org/

a few late-breaking slots still open, deadline: early September

Tutorials taught by, among others, some of the specialists present at Balisage:

- Michael Sperberg-McQueen
- Norman Walsh
- Andreas Witt

In Zadar, they eat shark. Really. And octopus ink risotto... They have grappa, too.

Don't miss on these goodies while they last! (BP is relocating to the Adriatic)

Thank you!

bansp at o2.pl

Inclusion semantics

Source text
(src.xml)

```
<source>
<w id="w1">word1</w>
<w id="w2">word2</w>
<w id="w3">word3</w>
<w id="w4">word4</w>
<w id="w5">word5</w>
<w id="w6">word6</w>
</source>
```

Annotation document

```
<result>
<s target="w1..w3" doc="src.xml"/>

<s target="w4..w6" doc="src.xml"/>
</result>
```

Result

```
<result>
<s target="w1..w3" doc="src.xml">
      <w id="w1">word1</w>
      <w id="w2">word2</w>
      <w id="w3">word3</w>
</s>
<s target="w4..w6" doc="src.xml">
      <w id="w4">word4</w>
      <w id="w5">word5</w>
      <w id="w6">word6</w>
</s>
</result>
```

Replacement semantics

Source text
(src.xml)

```
<source>
<w id="w1">word1</w>
<w id="w2">word2</w>
<w id="w3">word3</w>
<w id="w4">word4</w>
<w id="w5">word5</w>
<w id="w6">word6</w>
</source>
```

Annotation document

```
<result>
<repl target="w1..w2" doc="src.xml"/>
      <w id="w3">new content</w>
<repl target="w4..w6" doc="src.xml"/>
</result>
```

Result

```
<result>
      <w id="w1">word1</w>
      <w id="w2">word2</w>
      <w id="w3">new content</w>
      <w id="w4">word4</w>
      <w id="w5">word5</w>
      <w id="w6">word6</w>
</result>
```

Reverse Inclusion

Source text (src.xml)

```
<source>
<w id="w1">word1</w>
<w id="w2">word2</w>
<w id="w3">word3</w>
<w id="w4">word4</w>
<w id="w5">word5</w>
<w id="w6">word6</w>
</source>
```

Annotation document

```
<annotation>
<s target="w1..w3" doc="src.xml"/>

<s target="w4..w6" doc="src.xml"/>
</annotation>
```

("Virtual") result

```
<source>
<s>
    <w id="w1">word1</w>
    <w id="w2">word2</w>
    <w id="w3">word3</w>
</s>
<s>
    <w id="w4">word4</w>
    <w id="w5">word5</w>
    <w id="w6">word6</w>
</s>
</source>
```

Merger semantics

Annotation document

```
<annotation>
<seg lemma="x"
  target="a" doc="x.xml">
      <w>word1</w>
</seg>
<seg lemma="x"
  target="b" doc="x.xml">
      <w>word2</w>
</seg>
</annotation>
```

Annotation document

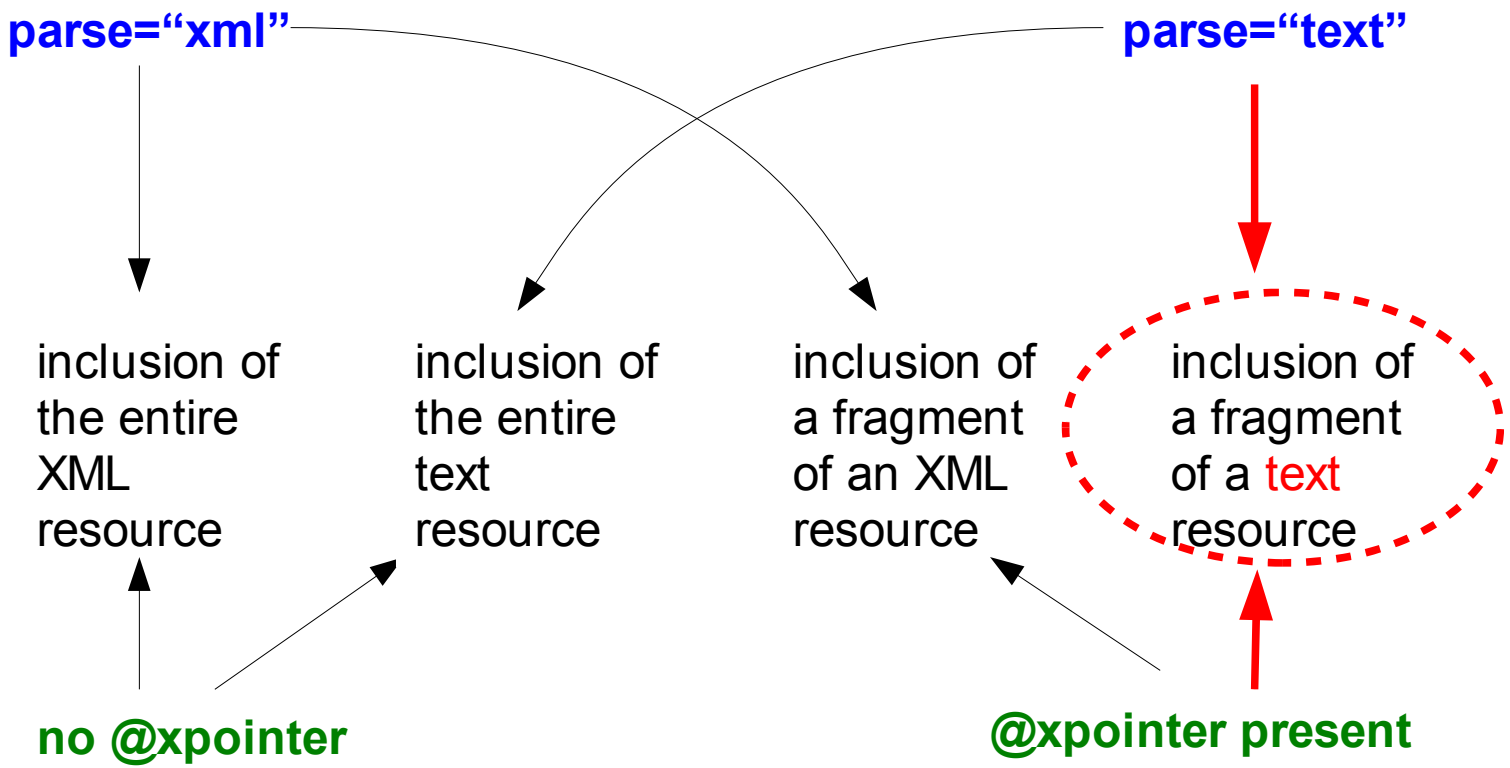
```
<annotation>
<seg pos="noun"
  target="a" doc="x.xml">
    <fs><!--feature structure--></fs>
</seg>
<seg pos="verb"
  target="b" doc="x.xml">
    <fs><!--feature structure--></fs>
</seg>
</annotation>
```

Result

```
<annotation>
<seg lemma="x" pos="noun"
  target="a" doc="x.xml">
    <w>word1</w>
    <fs><!--feature structure--></fs>
</seg>
<seg lemma="y" pos="verb"
  target="b" doc="x.xml">
    <w>word2</w>
    <fs><!--feature structure--></fs>
</seg>
</annotation>
```

XInclude

<xi:include href="resource.xml" parse="text|xml" xpointer="fragment_identifier" />



See my poster (at the back) for more discussion on this.