Multiple Annotations and XConcur

Andreas Witt

Tübingen University

In collaboration with

Oliver Schonefeld

Bielefeld University

Introduction

- The Multiple Annotations Approach deals with two different problems
  a. Syntax: The problem of annotating overlapping structures
  b. Concepts: Potential problems when annotating documents according to different, possibly heterogeneous tag sets
- Most often only the first aspect is addressed (see the title of our workshop)
- But both aspects belong together, in a way the aspect of syntax is less important
- But for the user confronted with practical tasks syntax is predominant

Background

- TEI (P3, P4, P5)
- Meaning and interpretation of concurrent markup, ALLC/ACH 2002
- Multiple hierarchies: new aspects of an old solution, Extreme Markup Languages, 2004
- Unification of XML Documents with Concurrent Markup. In: Literary and Linguistic Computing 20(1), 2005.
- Making CONCUR work. Extreme Markup Languages, 2005
- Towards validation of concurrent markup. Extreme, 2006

Background

- TEI (P3, P4, P5)
- Meaning and interpretation of concurrent markup, ALLC/ACH 2002
- *Multiple hierarchies: new aspects of an **old solution**, Extreme Markup Languages, 2004*
- Unification of XML Documents with Concurrent Markup. In: Literary and Linguistic Computing 20(1), 2005.
- Making CONCUR work. Extreme Markup Languages, 2005
- Towards validation of concurrent markup. Extreme, 2006

The “old solution”

An obvious and also simple solution would be to make a separate file for each transcription. …

Haugen (2004). Parallel Views: Multi-level Encoding of Medieval Nordic Primary Sources. In: *Literary and Linguistic Computing.* (19.1)

… However, this makes comparison between levels unnecessarily cumbersome, and it is notoriously difficult to keep track of revisions in parallel files.

Solutions mentioned by the TEI (P3,P4,P5)

- CONCUR: an optional feature of SGML that allows multiple hierarchies to be marked up concurrently in the same document
- milestone elements: empty elements that mark the boundaries between elements in a non-nesting structure
- fragmentation of an item: the division of a single element into two or more parts, each of which nests properly within its context
- virtual joins: the re-creation of a virtual element from fragments of text
- redundant encoding: information encoded in multiple forms

Problems with milestones

- milestones are empty elements

→ milestones elements have no content

- consequences:
  - no content model restriction can be stated by a document grammar
  - standard XML editors cannot annotate these regions
  - XML parsers cannot ensure proper nesting of the milestone elements
  - to process these regions by means of a style sheet is
    - more difficult (XSLT) or
    - impossible (CSS)

Problems with the other TEI-solutions

- fragmentation of an item:
  - ➢ results in 'containers' containing only a part of the text, e.g. a fragmented **`sentence`** or **`para`** would not contain an entire sentence or paragraph, as implied
- virtual joins:
  - ➢ requires a separate interpretation of the XML document
- redundant encoding:
  - ➢ results in multiple files

  → the files are not integrated in a larger unit

  → there exists no unit containing all the information

Redundant Encoding revisited

- rarely used by the markup community
- advantages (some are not unique to this approach):
  - each document is an independent unit of information
  - each level can be viewed separately
  - new levels can be added at any time, without reference to and dependence on existing files
  - standardized document grammars can be used and specialized document grammars can be defined in an intuitive way, e.g. some elements have text content, others not
  - modeling of alternative annotations based on different theoretical assumptions is possible
  - each document instance uses its own DTD (or Schema)
- main problem: multiple encodings in different forms are independent of each other

Multiple Annotations and their representation

- if the text content of the multiple annotations is identical the text can serve as the link of the multiple forms
- representations of this information can be created
- the ***representations*** make use of stand-off techniques
- two different representations have been adapted:
  - a Prolog-based representation originally developed by Michael Sperberg-McQueen, Claus Huitfeldt, and Allen Renear
  - an XML-based representation developed by Jean Carletta, Jonathan Kilgour, Tim O'Donnell, Stefan Evert, and Holger Voormann (see paper in the proceedings)

XML-based multi-layer annotation (example from TEI P5)

An encoding of a stanza of a poem
with respect to the metrical view

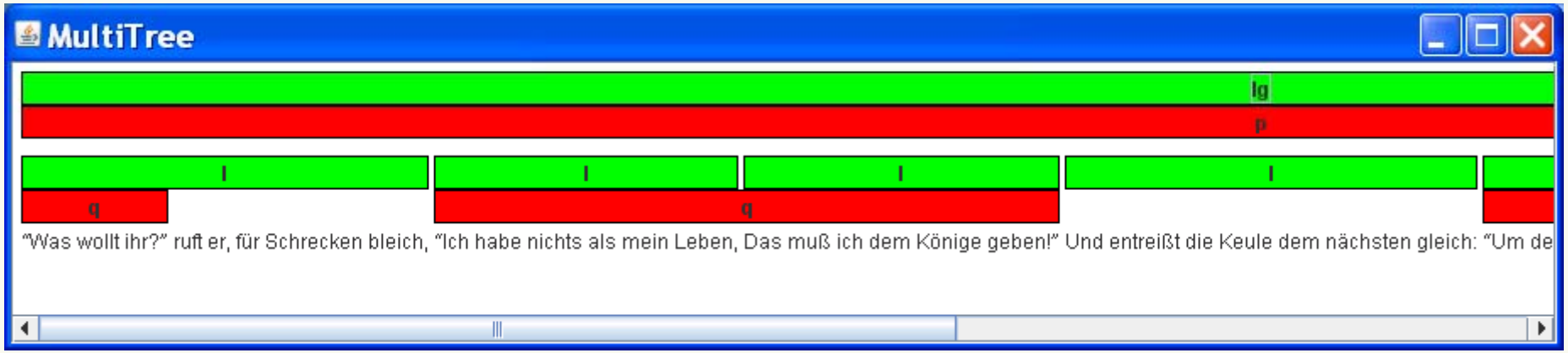
```
<lg type="stanza">
    <l>“Was wollt ihr?” ruft er, für Schrecken bleich,</l>
    <l>“Ich habe nichts als mein Leben,</l>
    <l>Das muß ich dem Könige geben!”</l>
    <l>Und entreißt die Keule dem nächsten gleich:</l>
    <l>“Um des Freundes willen erbarmet euch!”</l>
    <l>Und drei mit gewaltigen Streichen</l>
    <l>Erlegt er, die andern entweichen.</l>
</lg>
```

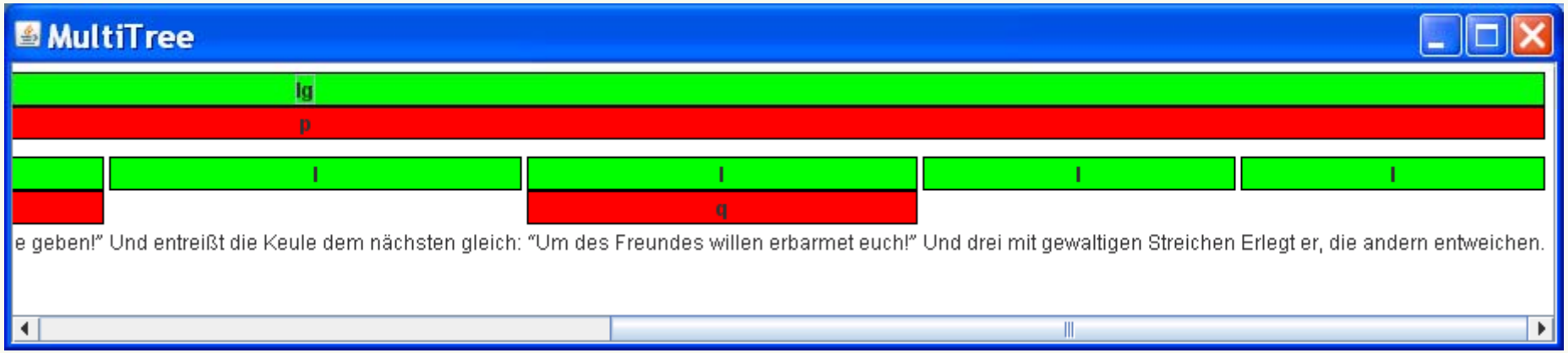
XML-based multi-layer annotation (example from TEI P5)

An encoding of a passage of a poem
with respect to the vocal view

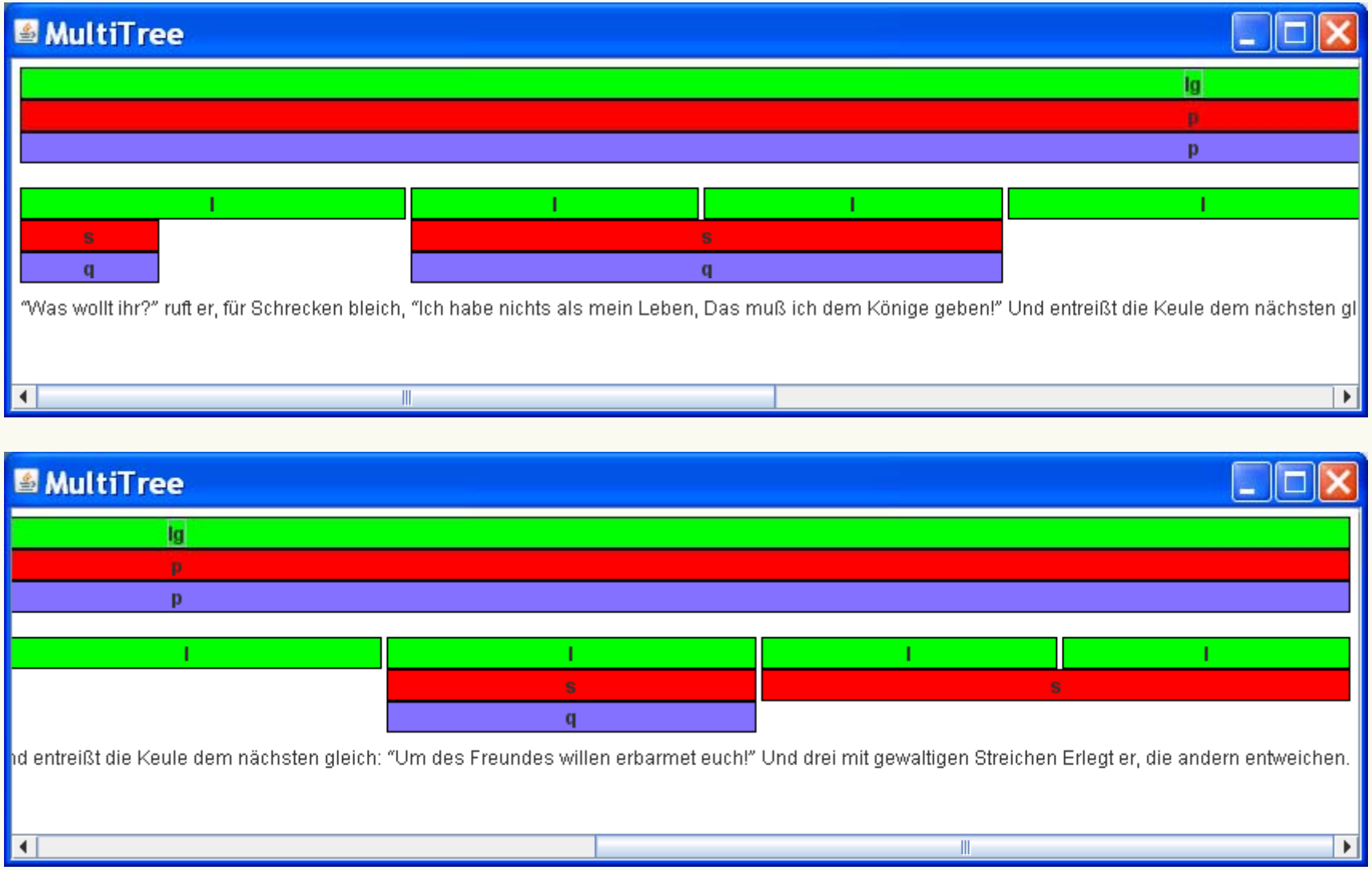
```
<p>
    <q>“Was wollt ihr?”</q> ruft er, für Schrecken bleich,
    <q>“Ich habe nichts als mein Leben,
    Das muß ich dem Könige geben!”</q>
    Und entreißt die Keule dem nächsten gleich:
    <q>“Um des Freundes willen erbarmet euch!”</q>
    Und drei mit gewaltigen Streichen
    Erlegt er, die andern entweichen.
</p>
```

Graphical view of the concurrent markup

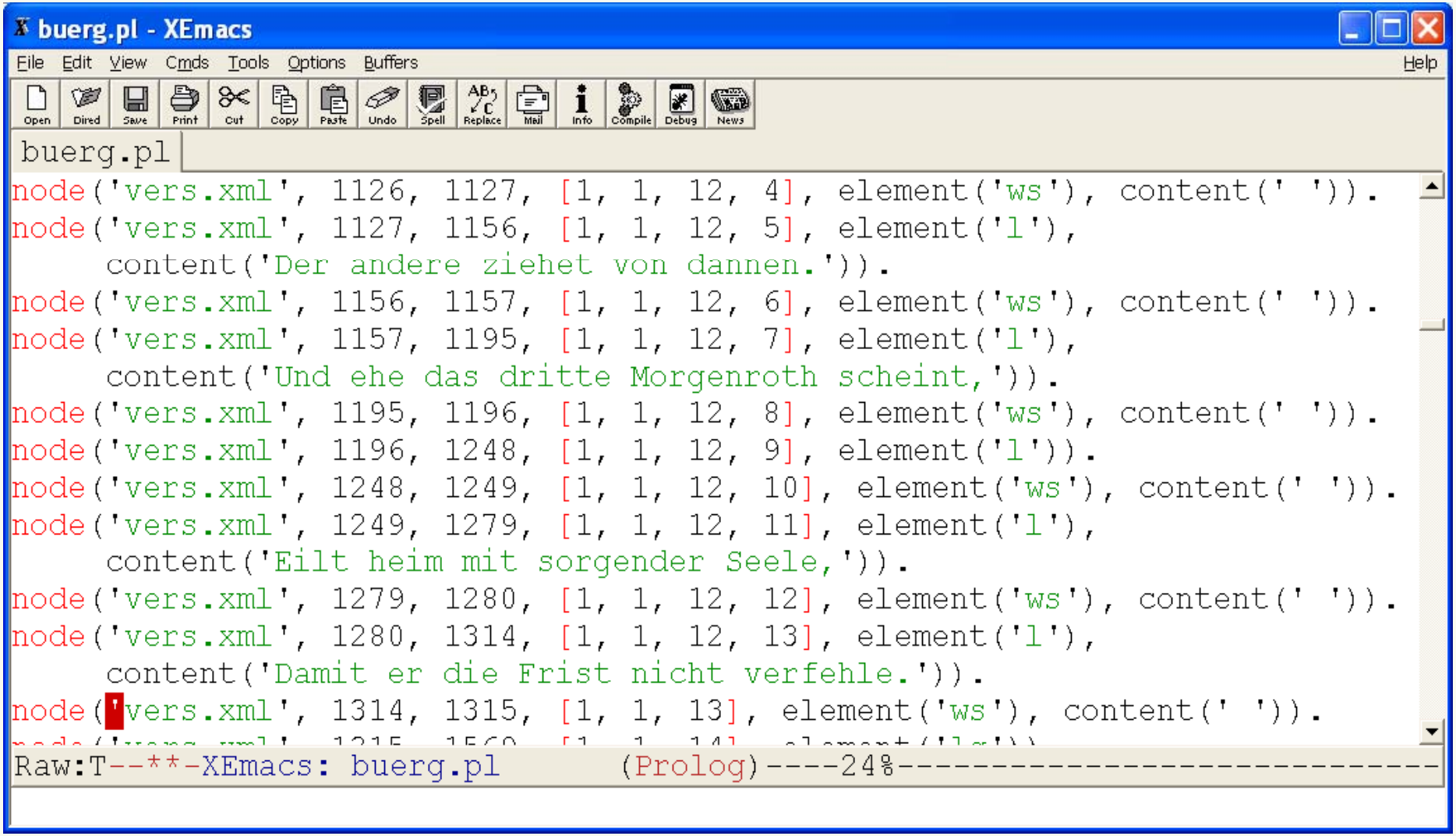




Graphical view of the concurrent markup



Prolog representation



XConcur

- Multiple XML-conformant annotated documents can be represented as one single marked-up document
- The resulting document might or not be a single hierarchy
- all elements are prefixed with an Annotation Layer Id and thereby assigned to an annotation layer
- An annotation schema for an annotation layer is declared
  - explicitly by using an Annotation Schema Declaration
  - implicitly by the markup used on the annotation layer
- XConcur is heavily influenced by SGML-Concur

XConcur vs. SGML Concur

- Most important differences to SGML
  - elements without an annotation layer id are not allowed
  - elements with the same generic identifier must be annotated explicitly
  - Examples:
  - XCONCUR and SGML CONCUR conformant notation:

```
<(l1)a><(l2)x>foo<(l1)br/><(l2)br/>bar</(l1)a></(l2)x>
```

  - SGML CONCUR conformant short-notation, not accepted by XCONCUR:

```
<(l1)a><(l2)x>foo<br/>bar</(l1)a></(l2)x>
```

XConcur vs. XML Namespaces

- XML Namespaces and annotation layer id are different concepts
  - XML Namespaces allow to use elements from different annotation schemas with conflicting names; elements must be nested properly
  - Annotation layer ids assign elements to an annotation layer
  - XCONCUR allows to use (potentially multiple) Namespaces on each annotation layer

Example

```
<?xconcur version="1.1" encoding="utf-8"?>

<(l1)lg type="stanza"><(l2)p>

  <(l1)l><(l2)q>“Was wollt ihr?”</(l2)q> ruft er, für
  Schrecken bleich,</(l1)l>

  <(l1)l><(l2)q>“Ich habe nichts als mein Leben,</(l1)l>

  <(l1)l>Das muß ich dem Könige geben!”</(l2)q></(l1)l>

  <(l1)l>Und entreißt die Keule dem nächsten
  gleich:</(l1)l>

  <(l1)l><(l2)q>“Um des Freundes willen erbarmet
  euch!”</(l2)q>
  </(l1)l>

  <(l1)l>Und drei mit gewaltigen Streichen</(l1)l>

  <(l1)l>Erlegt er, die andern entweichen.</(l1)l>

</(l2)p></(l1)lg>
```

Example

```
<?xconcur version="1.1" encoding="utf-8"?>

<(l1)lg type="stanza"><(l2)p>

  <(l1)l><(l2)q>“Was wollt ihr?”</(l2)q> ruft er, für
  Schrecken bleich,</(l1)l>

  <(l1)l><(l2)q>“Ich habe nichts als mein Leben,</(l1)l>

  <(l1)l>Das muß ich dem Könige geben!”</(l2)q></(l1)l>

  <(l1)l>Und entreißt die Keule dem nächsten
  gleich:</(l1)l>

  <(l1)l><(l2)q>“Um des Freundes willen erbarmet
  euch!”</(l2)q>
  </(l1)l>

  <(l1)l>Und drei mit gewaltigen Streichen</(l1)l>

  <(l1)l>Erlegt er, die andern entweichen.</(l1)l>

</(l2)p></(l1)lg>
```

Conclusion

- Several proposed solutions for both problems have been discussed
- the most simple solution, i.e. the annotation of these multiple structures or hierarchies in multiple files, is a simple way to represent documents with overlapping markup
- XConcur allows for an integrated representation of documents annotated on different levels

Multiple Annotations and XConcur

Andreas Witt

Tübingen University

In collaboration with

Oliver Schonefeld

Bielefeld University