Erschienen in: Mehler, Alexander/Lobin, Henning (Hrsg.): Automatische Textanalyse. Systeme und Methoden zur Annotation und Analyse natürlichsprachlicher Texte. - Wiesbaden: VS Verlag, 2004. S. 39-54.

Linguistische Informationsmodellierung mit XML

Andreas Witt

1 Einführung

Die linguistische Informationsmodellierung ist in zunehmendem Maße mit dem Gebrauch von SGML-basierten Auszeichnungssprachen verknüpft. Aufgrund der stetig wachsenden Verbreitung von XML wird oft vergessen, dass das Konzept der Auszeichnungssprachen (oder im Englischen ‚Markup Languages') keineswegs auf die Sprachen der SGML-Familie beschränkt ist, da es grundsätzlich natürlich möglich ist, unterschiedlichste Formate zu definieren, die es erlauben, textuelle Daten mit zusätzlichen Informationen zu versehen bzw. auszuzeichnen.

Die Verwendung von Auszeichnungssprachen kann aus verschiedenen Perspektiven heraus betrachtet werden. Eine anwendungsorientierte Sicht fokussiert die Informationsanreicherung, die den Austausch und die Wiederverwendbarkeit der Daten vereinfacht. Für diese Zwecke stellt die Extensible Markup Language eine sehr gut geeignete Grundlage dar, und in dieser Anwendung liegt auch die große Popularität von XML begründet.

Hiervon ist eine theoretischere Sichtweise auf die Anwendung von XML zu unterscheiden: Die Informationsanreicherung mit Auszeichnungssprachen dient der Informationsmodellierung (vgl. Lobin 2000). Bei der theoretisch fundierten Modellierung komplexer Strukturen kann es sich ergeben, dass einige davon nicht direkt in XML – oder auch in deren allgemeineren Obermenge SGML – abbildbar sind. In dem vorliegenden Beitrag sollen insbesondere die Aspekte der Informationsmodellierung diskutiert werden, die in den Grenzbereich der Möglichkeiten von XML hineinreichen.

Für die Modellierung linguistischer Informationen werden die SGML-basierten Auszeichnungssprachen immer häufiger als außerordentlich gut geeignet angesehen.[1] Allerdings zeigen sich gerade bei dieser Art von Information auch die Grenzen von XML. Dies ist nicht notwendigerweise ein Widerspruch, da beides damit zusammenhängt, dass linguistische Information äußert komplex sein kann.

---

[1] Einem Beitrag mit dem Titel „SGML und Linguistik" (Witt 1999) habe ich die Zeile ‚In einer Freundschaft wie dieser gibt es kein zurück' vorangestellt, um in einer pointierten Form auf die gute Eignung von SGML für die linguistische Informationsmodellierung hinzuweisen. Dieser Beitrag erschien in „Text im digitalen Medium" (Lobin 1999), einem Band, der in gewisser Weise als Vorgänger des vorliegenden Buches bezeichnet werden kann.

Wird XML hinsichtlich ihrer Eignung für die linguistische Informationsmodellierung betrachtet, fällt auf, dass XML im Wesentlichen nur eine Informationsebene modellieren kann, auch wenn diese Ebene häufig durchaus relativ komplex strukturiert ist. Die Ursache hierfür ist in der Geschichte des XML-Vorläufers SGML zu finden: Die Textauszeichnung diente nahezu ausschließlich als Grundlage für den Buchdruck, entsprechend stand die logische Strukturierung der Texte in Überschriften, Abschnitte, Fußnoten etc. im Zentrum der Verwendung von SGML. Für die Präparation von Manuskripten für den Buchdruck gab es nur selten die Notwendigkeit, mehr als diese Ebene der logischen Struktur zu betrachten.[2] Da XML eine Untermenge von SGML bildet, gelten die alten Beschränkungen nach wie vor.

Sollen mehrere Informationsebenen in einer generellen Form strukturiert werden, so ist es entweder notwendig innerhalb des XML-Paradigmas Behelfslösungen zu konstruieren oder es werden Auszeichnungssprachen definiert, die allgemeiner sind als XML.

Im vorliegenden Artikel werden Facetten dieser Problematik aufgezeigt. Es werden XML-basierte und nicht-XML-basierte Lösungen diskutiert. Gegen Ende des Beitrages wird ein Ansatz vorgestellt, der es erlaubt, XML in gewohnter Weise zu verwenden und dennoch unterschiedliche, z.T. inkompatible Informationsebenen zu modellieren.

2 Multiple Hierarchien, Multidimensionalität und Überlappungen

Texte können mit Annotationen versehen werden. Hierdurch werden Informationen expliziert, die im eigentlichen Text nicht enthalten sind. Die Art dieser zusätzlichen Informationen unterliegt keinerlei thematischen Beschränkungen. So kann z.B. ein Text aufbereitet werden, um die Ergebnisse einer literaturwissenschaftlichen Analyse in den Annotationen zum Ausdruck zu bringen. Eine derartige Aufbereitung unterscheidet sich nahezu vollständig von einer linguistischen Annotation.

Bei genauerer Betrachtung fällt auf, dass es natürlich nicht jeweils genau eine literaturwissenschaftliche bzw. linguistische Informationsanreicherung gibt, sondern eine schier unbegrenzte Vielzahl von Annotationsmöglichkeiten. So spiegelt sich in einer literaturwissenschaftlich motivierten Annotation zweifelsohne eine Theorie wider und mit dem Wechsel der zugrunde liegenden Theorie wechselt denn auch die Informationsanreicherung. Noch deutlicher wird dieses Problem bei der linguistisch motivierten Informationsmodellierung, weil

---

[2] Einer dieser seltenen Fälle tritt dann ein, wenn genauere Information über das Layout des Textes in das Manuskript eingefügt werden soll. Ein vieldiskutiertes Beispiel hierfür ist die Einfügung von Paginierungsinformationen in das Manuskript. Hierfür stellt SGML eine Methode (CONCUR) zur Verfügung, welche de facto jedoch in SGML nie implementiert wurde und von deren Verwendung bereits im kommentierten SGML-Standard (Goldfarb 1990) abgeraten wird.

ihr Gegenstand völlig unterschiedliche Ebenen der Sprachbeschreibung (z.B. Syntax, Semantik, Phonologie etc.) sein können. Da ein Text meist aus nur einem spezifischen Untersuchungsinteresse heraus betrachtet und dieser Text auch nur bezüglich einer Ebene annotiert wird, wird diese Einschränkung für viele Verwendungsweisen von XML übersehen. Insbesondere für die Anreicherung der Texte mit linguistischer Information ist diese Beschränkung auf eine Annotationsebene jedoch zu restriktiv.

Gary F. Simons beschreibt in einem Artikel *the nature of linguistic data.* (Simons 1998). Neben verschiedenen weiteren Kriterien wird darin hervorgehoben, dass Sprachdaten nicht nur hierarchisch geordnet, sondern – darüber hinaus – auch multidimensional strukturiert sind. Simons betont einerseits die gute Eignung von SGML für die Modellierung von Hierarchien, andererseits hebt er hervor, dass SGML nur in eingeschränktem Maße geeignet ist, die Multidimensionalität von Sprachdaten zu repräsentieren.

Formal betrachtet erlaubt SGML (und mithin ihre Untermenge XML) bezogen auf die Strukturierung eines gegebenen Dokuments die Repräsentation genau einer Hierarchie. Infolgedessen kann im Prinzip auch nur eine Struktur repräsentiert werden. In der Praxis ergeben sich durch diese Restriktion relativ selten Probleme, da verschiedene Strukturen häufig in einer Hierarchie repräsentiert werden können. So unterscheidet sich z.B. die so genannte logische Struktur eines Textes, d.h. die Einteilung in Überschriften, Aufzählungen, Abschnitte etc., vollständig von einer syntaktischen Struktur, z.B. die Einteilung des Textes in Sätze und Phrasen. Nichtsdestotrotz ist es meist problemlos möglich, diese beiden Strukturen in einer Hierarchie abzubilden. Dies führt allerdings notwendigerweise zu einer Vermischung der Ebenen, sowohl auf der Ebene der annotierten Texte als auch auf der Ebene der Dokumentgrammatiken.[3]

Durch Entwicklungen im Bereich von XML können heute auch Unterscheidungen dieser Strukturen vorgenommen werden. Dies kann durch den Gebrauch der so genannten Namensräume (Namespaces, vgl. Bray *et al.* 1999) geschehen. Hierbei können die verschiedenen Strukturierungen durch unterschiedliche Dokumentgrammatiken lizenziert werden. Die Verknüpfung der Elemente mit den sie definierenden Dokumentgrammatiken erfolgt, indem die Elementnamen mit einem Präfix versehen werden, das auf die Strukturdefinition verweist.[4] So kann die logische Struktur des Textes z.B. mit den (X)HTML-Elementen für Überschriften, Abschnitte, Listen etc. ausgezeichnet werden und die syntakti-

---

[3] Die Struktur eines Dokumenttyps wird durch eine Dokumentgrammatik festgelegt. Eine Dokumentgrammatik wird ihrerseits in einer speziellen Sprache, einer Schemasprache, formuliert. Die bekannteste Schemasprache ist die im XML-Standard definierte Sprache zu Ausdruck von XML-DTDs.

[4] Leider erlauben nicht alle Schemasprachen die Verwendung dieses Mechanismus. Insbesondere die XML-DTDs sehen keine Behandlung von Namensräumen vor.

sche Struktur durch die Verwendung des entsprechenden Moduls der TEI-DTD. Werden die entsprechenden Namensräume definiert, kann z.B. auf eine aus der logischen Textstruktur stammende Überschrift mit `<html:h2>` statt schlicht mit `<h2>` verwiesen werden, während ein Wort oder ein Morph mit `<tei:w>` bzw. `<tei:m>` statt mit `<w>` bzw. `<m>` markiert wird. Eine relativ detaillierte Annotation eines Textes kann dann z.B. die folgende Passage enthalten:

```
<html:h2><tei:w><tei:m>Ein</tei:m><tei:m>leit</tei:m>
<tei:m>ung</tei:m></tei:w></html:h2>
```

Diese Anreicherung der Annotationen vereinfacht es, den Bezug der Annotationen zu bestimmten Ebenen (hier Textstruktur und Morphologie) zu erkennen.

Die durch die Verwendung von Namensräumen ermöglichten differenzierteren Strukturierungen reichen jedoch noch nicht aus, der Multidimensionalität von Sprache vollständig gerecht zu werden. So gibt es zum Teil in ein und demselben Annotationsformat in Abhängigkeit der zugrunde gelegten Theorien verschiedene Annotationsalternativen. Sollen alle Möglichkeiten uniform ausgezeichnet werden, führt dies zu Problemen. Eine naheliegende, aber inkorrekte Lösungsmöglichkeit bestünde darin, unterschiedliche Namensraumpräfixe zu verwenden, um die Alternativen zu annotieren. So könnte ein Präfix `tei_a1` und ein zweites Präfix `tei_a2` definiert werden, die beide auf die TEI-DTD verweisen. Da die Präfixe allerdings nur eine Platzhalterfunktion für die expandierten Namensräume besitzen, gibt es keinen Unterschied zwischen `tei_a1` und `tei_a2` – es sei denn sie verwiesen auf unterschiedliche Dokumentgrammatiken.[5, 6]

Ein weiteres Problem der Mehrdimensionalität ist sehr eng mit dem XML-Standard verwoben und daher fundamentaler Natur: Durch Elemente ausgezeichnete Textteile dürfen sich nicht überlappen![7] Das gilt – bis auf den in Fußnote 2 angesprochenen Spezialfall – für SGML, es gilt sowohl für gültige als auch für wohlgeformte XML-Dokumente, es gilt aber auch für die „laxeste" Form der Definition von XML, d.h. für so genannte XML-Fragmente. Dieser Sachverhalt trifft unabhängig von der Verwendung von Namensräumen zu.

Die Überlappungsproblematik soll anhand der Erweiterung des obigen Beispiels veranschaulicht werden. Soll der Überschriftentext ‚Einleitung' nicht nur

---

5 Es wird derzeit diskutiert, ob der definierte Namensraum tatsächlich mit einer Dokumentgrammatik verbunden sein muss. Dies ist in der Praxis derzeit nicht der Fall.

6 Zwar könnten Kopien der Dokumentgrammatiken angefertigt werden, die unterschiedlichen Namensräumen zugeordnet wären. Dies wäre allerdings ein klarer Verstoß gegen den Geist des Namensraum-Standards, da Namensräume genau deshalb verwendet werden, um auf standardisierte Dokumentgrammatiken in eindeutiger Form Bezug nehmen zu können.

7 Die durch SGML und XML ausgezeichneten Einheiten bilden eine 'Ordered Hierarchy of Content Objects' (OHCO). Zur kritischen Diskussion vgl. Renear *et al.* (1996).

morphologisch sondern auch entsprechend seiner Silbenstruktur annotiert werden, so würden sich die ausgezeichneten Umgebungen überlappen:

```
*   <w><m><syll>Ein</syll></m><m><syll>
    lei</syll><syll>t</m><m>ung</m></syll></w>
```

Das Überlappungsproblem tritt eben wegen der Multidimensionalität von natürlicher Sprache immer wieder auf. Die Gültigkeit dieser Feststellung bedeutet jedoch nicht, dass die linguistischen Beschreibungsebenen selbst unabhängig voneinander sind – es sind ja gerade die Wechselwirkungen dieser Ebenen für immer mehr Linguist(inn)en interessant. Allerdings hat das Auftreten derartig komplexer Strukturen immer wieder zu einer Infragestellung der SGML-basierten Markup-Sprachen geführt:

> The most persistent complaint of SGML's critics among humanists is that SGML simply *cannot* handle such overlapping features. In the general form stated, this claim is untrue, but it is fair to say that handling overlap requires some substantial extensions to what is otherwise a rather simple data model.
> Sperberg-McQueen/Huitfeldt (1999: 29f., Hervorhebung beibehalten)

Eine der möglichen Lösungen besteht darin, derartige Überlappungen zuzulassen, d.h. es kann eine neue Auszeichnungssprache entwickelt werden, die die Überlappungsrestriktion aufgibt. Ein Beispiel für eine derartige Markup-Sprache ist das Multi-Element Code System (MECS) bzw. TexMECS (Huitfeldt/ Sperberg-McQueen 2001).

Eine Möglichkeit, Überlappungen in XML auszudrücken, besteht darin, dass an den Stellen im Markup, an denen sich Konflikte ergeben, die betreffenden Umgebungen pro forma geschlossen und anschließend wieder geöffnet werden. Ein spezielles Attribut erlaubt die Buchhaltung darüber. Das inkorrekte Beispiel könnte dann folgendermaßen verändert werden:

```
<w><m><syll>Ein</syll></m><m><syll>lei</syll><syll
  stat="i" id='s1' next='s2'>t</syll></m><syll
  stat="i" id='s2' prev='s1'><m>ung</m></syll></w>
```

Der Wert `i` des Attributs `stat` gibt an, dass das Element unvollständig (bzw. *incomplete*) ist. In den Werten der Attribute `prev` und `next` wird auf die fehlenden Teile verwiesen. Werden die Attribute nicht angegeben, können die Attribute Defaultwerte annehmen.

Durch Verwendung der Fragmentierungstechnik ist es also, wenn auch auf relativ umständliche Weise, durchaus möglich, überlappende Umgebungen im XML-Format zu modellieren, d.h. eine Aufgabe der Überlappungsrestriktion ist demnach nicht unbedingt notwendig. Der von TexMECS beschrittene Weg bietet jedoch zweifelsohne die naheliegendere Modellierung, wenn auch, in formaler Hinsicht, die komplexere. Beide Methoden bergen aber einen fundamentalen Nachteil, der sich auch im Zuge der ‚normalen' Verwendung von XML ergibt: Die Annotationen sind sequentialisiert, d.h. auch wenn durch geschickte

Namenswahl oder durch den Gebrauch von Namensräumen auf eine Ebenenzugehörigkeit hingewiesen wird und auch wenn die Umgebungen frei ineinander verschachtelt sein können, so werden doch immer die verschiedenen Umgebungen vor oder nach anderen Umgebungen geschlossen oder geöffnet.

3 Beziehungen zwischen Annotationsebenen

Zur Verdeutlichung des Problems der Ebenenzugehörigkeit soll erneut das Wort ‚Einleitung' lexikalisch und morphologisch annotiert werden:

```
<w><m>Ein</m><m>leit</m><m>ung</m></w>
```

Diese Annotation ist naheliegend und findet sich in diversen linguistischen Dokumentgrammatiken.[8] Eine naive Verbalisierung dieser Struktur könnte wie folgt lauten: „Das Wort beginnt, dann folgen drei Morphe, dann wird das Wort geschlossen". Wenn diesem Wort ein definiter Artikel vorangestellt wäre und dieser annotiert werden sollte, würde dieser analog als `<w><m>die</m><w>` ausgezeichnet. Eine mögliche Verbalisierung der Information wäre: „Das Wort enthält ein Morph". Dies ist allerdings eine Interpretation, die keineswegs selbstverständlich ist, vielmehr beginnen und enden Wort und Morph gleichzeitig. Sowohl in der Sequenz von Auszeichnungen und dem ausgezeichneten Text ‚die' als auch in der Baumrepräsentation umschließt allerdings das Element `w` das Element `m`.

Durusau/O'Donnell (2002) stellen eine Liste von verschiedenen Verbindungen und gegenseitigen Abhängigkeiten von XML-annotierten Umgebungen zusammen, die von ihnen als partielle Typologie sich überlappender Hierarchien bezeichnet wird.[9] Die dort aufgeführten 13 Fälle werden nachfolgend zu 7 Fällen zusammengefasst[10] und an Beispielen aus der Linguistik exemplifiziert.

1. Das Umgebungsende der einen Ebene ist identisch mit einem Umgebungsbeginn der anderen Ebene.

```
<a>.................</a>
                      <b >.........</b>
```

Als Beispiel für diesen Fall kann die Annotation einer syllabischen und die Auszeichnung einer phonetischen Ebene herangezogen werden: Pausen in gesprochenen Wörtern beginnen oft mit dem Ende von Silben.

---

8 Als ein Beispiel sei hier nur die DTD der *Text Encoding Initiative* aufgeführt.

9 Die Hierarchie basiert ihrerseits u.a. auf Durand (1999).

10 Die Differenz ergibt sich daraus, dass Durusau/O'Donnell die Fälle 1-6 jeweils ‚doppelt' aufgeführt haben, d.h. bestimmte Fälle von Durusau/O'Donnell können ohne Beschränkung der Allgemeinheit paarweise zusammengefasst werden. Z.B. unterschieden sie die Fälle „`a` beginnt und endet vor `b`" und „`b` beginnt und endet vor `a`".

2. Die annotierten Umgebungen überlappen sich.

```
<a>.....................</a>
                    <b>.............................</b>
```

Die bereits angesprochene Überlappung von Silben und Morphemen ist ein Beispiel für diesen Fall.

3. Die Enden der ausgezeichneten Umgebungen sind identisch, die Anfänge der Umgebungen nicht.

```
                  <a>..................... </a>
<b>.................................</b>
```

Ein Beispiel für diesen Fall bildet die Annotation von Wörtern und die Auszeichnung von Affixtypen. Werden diese Ebenen markiert, wird das (letzte) Suffix am Ende des Wortes geschlossen.

4. Eine Umgebung ist vollständig in der anderen Umgebung enthalten.

```
          <a>.....................</a>
<b>.....................................</b>
```

Die in einigen Sprachen vorkommenden Infixe sind in Wörtern enthalten.

5. Die Anfänge der ausgezeichneten Umgebungen sind identisch, die Enden nicht.

```
<a>.....................</a>
<b>......</b>
```

Werden Wortarten und Sätze in deutschen Texten detailliert ausgezeichnet, sind Startpunkt des Interrogativpronomens und der Beginn eines Fragesatzes identisch.

6. Die annotierten Umgebungen sind identisch.

```
<a>.....................</a>
<b>.....................</b>
```

Beispielsweise sind in monomorphematischen Wörtern Morphem- und Wortgrenzen identisch.

7. Es werden unterschiedliche Umgebungen annotiert die sich nicht berühren (dieser Fall ist eigentlich ein ‚Nicht-Fall').

```
                              <a>.................</a>
<b>......</b>
```

Die gemäß zwei Annotationsebenen ausgezeichneten Einheiten überlappen nicht.

Eine derartige Notation fokussiert Beziehungen zwischen verschiedenen Ebenen. Die meisten dieser Beziehungen (und zwar alle außer der unter Punkt 3 aufgeführten ‚klassischen' Überlappung) existieren jedoch im Prinzip auch bei der Annotation einer Hierarchieebene; sie bleiben jedoch dort oft implizit.

4 Modellierung unterschiedlicher Annotationsebenen

4.1 Klassische Ansätze

Auch innerhalb der etablierten Formalismen SGML und XML ist es möglich, eine Repräsentation zu verwenden, in der überlappende Einheiten modelliert werden können. Eine dieser Möglichkeiten bildet die bereits oben angesprochene Fragmentierungstechnik. In Kapitel 31 der TEI-Guidelines (Sperberg-McQueen/Burnard 1994) wie auch von Barnard *et al.* (1995) werden neben diesem Ansatz weitere Techniken zusammengefasst und beschrieben. Es ist gezeigt worden, dass die Fragmentierungstechnik eine Möglichkeit bietet, eine Informationsmodellierung vorzunehmen, die sehr nah an der von TexMECS erlaubten, in XML jedoch unzulässigen, direkten Überlappung von Elementgrenzen (Fall 2 in Abschnitt 3) liegt. Bei diesen Ansätzen wird der Informationsmodellierung innerhalb einer Ebene besondere Bedeutung beigemessen. Sollen hingegen die Beziehungen zwischen den Informationsebenen ausgedrückt werden, bieten sich andere ‚klassische' Techniken zur Modellierung von Überlappungen innerhalb von SGML und XML an. Die wichtigsten hiervon sind:

1. Verweise bzw. Hyperlinks auf eine vorhandene primäre Annotation,
2. Hyperlinks auf eine Zeitachse und
3. Anker bzw. Meilensteine.

Diese Techniken seien im Kontext linguistischer Anwendungen kurz beschrieben.

Das europäische Großprojekt MATE[11] verwendet eine primäre Annotationsebene – in den Dokumentationen zu MATE wird hierfür meist die Wortebene genannt – und verknüpft diese mit den weiteren Ebenen. Soll z.B. die Präpositionalphrase aus der Schlagzeile „PDS: Ein Schritt nach vorn in den Abgrund"[12] annotiert werden erfolgt eine Primärannotation:

---

[11] Das Projekt MATE (Multi-level Annotation Tools Engineering, vgl. McKelvie *et al.* 2001, Bernsen *et al.* 2002) wurde 1999 beendet, findet aber im Projekt NITE (Natural Interactivity Tools Engineering, vgl. Soria *et al.* 2002) eine Fortsetzung. Die hier beschriebene XML-Annotationstechnik bezieht sich auf MATE.

[12] Aus: ‚die tageszeitung' vom 14. 10. 2002.

```
... <w id="w04">Schritt</w> <w id="w05">nach</w>
    <w id="w06">vorn</w> <w id="w07">in</w>
    <w id="w08">den</w> <w id="w09">Abgrund</w> ...
```

Mit dieser Ebene wird die weitere Ebene (im Beispiel das „chunking level") verknüpft:

```
<ch id="ch03" type="P" href="#id(w07)..id(w09)"/>
```

Diese Verknüpfung mit einer primären Annotationsebene kann mit der zweiten oben angegebenen Technik verglichen werden. Auch bei der Verknüpfung der Annotationen mit einer Zeitachse wird eine Ebene als Verknüpfungsbasis verwendet, mit der die anderen Ebenen in Beziehung gesetzt werden. Im Gegensatz zur Verknüpfung mit einer linguistischen Primärebene ist bei der Verwendung der Zeitachse als Verankerungsebene die Annotationsbasis nicht selbst Gegenstand der Informationsmodellierung. Diese Technik findet sich in den so genannten *annotation graphs* (AGs; Bird/Liberman 2001)[13].

Auch die Verwendung von Anker- bzw. Meilensteinelementen kann als Aufspaltung der Annotation aufgefasst werden. Eine Ebene enthält Textdaten, die mit zusätzlichen Auszeichnungen versehen wurden. Die weiteren Ebenen werden, ähnlich wie im MATE-Beispiel, durch Verknüpfungen mit dieser Ebene erreicht. Die Technik der Meilensteine steht also in enger Beziehung mit der Verwendung von Hyperlinks auf eine vorhandene primäre Annotation. Der Unterschied besteht darin, dass nicht eine existierende Ebene (z.B. die Wortebene) als Primärebene verwendet wird, sondern dass zusätzlich zu dem in der Basisebene verwendeten Markup bei Bedarf Verknüpfungsziele eingefügt werden. Hierfür werden leere Elemente – so genannte Meilensteine – benutzt. Auf diese wird dann verwiesen, wenn die primäre Ebene nicht fein genug unterteilt wurde, um allein als Verknüpfungsbasis für die weiteren Informationsebenen dienen zu können.

Mit diesen Techniken ist allerdings auch eine Reihe von Nachteilen verbunden. Neben praktischen Problemen, die zwangsläufig auftreten, wenn in die Grenzbereiche etablierter Formalismen vorgedrungen wird, sind hier vor allem theoretische Probleme der Informationsmodellierung zu nennen.

Die praktischen Probleme sind insbesondere mit den Nutzungsmöglichkeiten etablierter Software verbunden, was dazu führt, dass nicht nur die Informationsstrukturierung erschwert wird, sondern auch die automatische oder halbautomatische Manipulation (sprachlicher) Daten.[14]

---

[13] Die Verknüpfung der Annotation mit einem Zeitstrahl wurde ursprünglich bereits in den oben erwähnten Publikationen der Text Encoding Initative (TEI) vorgestellt. Die TEI-DTD definiert hierfür das Element <timeline> (vgl. Sperberg-McQueen/Burnard 1994, S. 438 ff.).

[14] Werden XML-Dateien erfasst oder weiterverarbeitet, kommen z.B. XML-Parser zum Einsatz. Mit ihnen wird die Korrektheit der XML-Dateien bezüglich der in der Dokumentgrammatik ausgedrückten Struktur überprüft. Dies funktioniert allerdings nur bezüglich der primären Ebe-

Auf der theoretischen Seite stellt sich u.a. das Problem, dass durch die Einführung einer primären Annotationsebene die linguistischen Ebenen nicht gleichberechtigt sind.

4.2 Allgemeinere Auszeichnungssprachen

Wie bereits am Beispiel von TexMECS gesehen, ist die Verwendung von SGML-basierten Auszeichnungssprachen kein unumstößliches Dogma. TexMECS ist ein Beispiel für eine allgemeinere Auszeichnungssprache. Ihre wesentliche Erweiterung betrifft die angesprochene Zulassung sich überlappender Umgebungen. Im Jahr 2002 wurde mit der Definition einer weiteren Markup-Sprache begonnen. Diese wird als „Layered Markup and Annotation Language (LMNL)" bezeichnet (Tennison/Piez 2002). LMNL ist eine Markup-Sprache, die es nicht nur erlaubt, sich überlappende Elemente zu annotieren, sondern darüber hinaus die Elementnamen mit bestimmten Annotationsebenen zu verbinden.

Alle im XML-Format modellierbaren Strukturen können auch mit LMNL modelliert werden. Als Beispiel für die Notation von LMNL soll die morphologische Struktur des Wortes „Einleitung" in XML-Syntax und als LMNL-Dokument dienen:

XML: `<w><m>Ein</m><m>leit</m><m>ung</m></w>`
LMNL: `[w}[m}Ein{m][m}leit{m][m}ung{m]{w]`

Soll eine Mehrebenen-Annotation erfolgen, so wird dies in LMNL durch die Definition sogenannter ‚Layers' erreicht. Diese Ebenen können ihrerseits auf anderen Ebenen oder auf den textuellen Daten basieren. Sollen z.B. eine phonologische und eine morphosyntaktische Annotationsebene verwendet werden, können diese folgendermaßen definiert werden:

```
[!layer name="phon" base="#text"]
[!layer name="mosy" base="#text"]
```

Auf diese Ebenendefinition kann bei der Auszeichnung der zu annotierenden Bereiche Bezug genommen werden. Verbunden mit der Aufhebung der Überlappungsrestriktion kann die syllabische und morphologische Struktur des Beispielwortes folgendermaßen markiert werden:[15]

```
[w~mosy}
  [m~mosy}[syll~phon}Ein{syll~phon]{m~mosy]
  [m~mosy}[syll~phon}lei{syll~phon]
          [syll~phon}t{m~mosy]
```

---

ne; ist die primäre Ebene ein Zeitstrahl, so sind derartige Überprüfungen überhaupt nicht möglich.

[15] Die Einrückungen sind kein Bestandteil der Annotation, sie sollen der Lesbarkeit dienen.

```
                [m~mosy}ung{m~mosy]{syll~phon]
{w~mosy]
```

Dieses Beispiel zeigt, dass es LMNL erlaubt, Strukturen zu modellieren, die für SGML-basierte Markup-Sprachen ein notorisches Problem darstellen. Darüber hinaus bietet LMNL die Möglichkeit, Zusatzinformationen, die in XML üblicherweise in Attributen ausgedrückt werden, in einer strukturierten Weise auszudrücken. Konkret bedeutet dies, dass das LMNL-Analogon zu den XML-Attributwerten nicht nur Zeichenketten beinhalten darf, sondern strukturierte Dokumente. Für die linguistische Informationsmodellierung bedeutet dies u.a., dass mit LMNL ein Formalismus zur Verfügung steht, der die in der (Computer-)Linguistik verwendeten Attribut-Wert-Strukturen in nahe liegender Weise auszudrücken erlaubt.[16]

Zusammenfassend kann gesagt werden, dass die *Layered Markup and Annotation Language* das Potenzial besitzt, für die (linguistische) Informationsmodellierung eine wichtige Rolle zu spielen, da sie eine Reihe der SGML-inhärenten Restriktionen aufhebt. Der Preis dafür ist jedoch hoch: Ein etablierter und für die maschinelle Verarbeitung hervorragend geeigneter Formalismus würde aufgegeben!

4.3 Multiple Annotation

Als Alternative zu den bisher vorgestellten ‚klassischen' Ansätzen der Modellierung unterschiedlicher Annotationsebenen auf der einen Seite und der Verwendung allgemeinerer Markup-Sprachen auf der anderen Seite kann die separate Annotation der verschiedenen Informationsebenen angesehen werden (vgl. Witt 2002). Hierbei wird dieselbe textuelle Datenbasis entsprechend der zu annotierenden Ebenen vervielfältigt und separat annotiert.[17]

Der Vorteil der separaten multiplen Annotation derselben Daten besteht u.a. darin, dass hierdurch die Informationsmodellierung einer Ebene nicht von (der Existenz) einer anderen modellierten Ebene abhängig ist. Dies führt dazu, dass jede Ebene separat betrachtet werden kann und jederzeit neue Ebenen hinzugefügt werden können. Auch ist die Modellierung alternativer Annotationen möglich, die auf unterschiedlichen theoretischen Grundannahmen basieren.[18] Dieser Ansatz soll anhand eines Beispiels dargestellt werden.

---

[16] Die Modellierung von Merkmalsstrukturen in SGML erscheint bei oberflächlicher Betrachtung zunächst eine triviale Aufgabe zu sein. Genauere Betrachtungen der Problematik zeigen allerdings, dass umfassende und adäquate Lösungsalternativen dieser Aufgabe sehr komplex sein können (vgl. Simons/Langendoen 1995 und Witt 2002).

[17] Technisch kann auf den Vervielfältigungsschritt verzichtet werden, da dieselbe Datengrundlage durch das so genannte Stand-Off-Markup annotiert werden kann (Thompson/McKelvie 1997).

[18] Z.B. zeigen Asahara *et al.* (2002), dass die Bestimmung der Wortgrenzen im Japanischen umstritten ist und dass u. U. die verschiedenen Alternativen annotiert werden müssen. Asahara

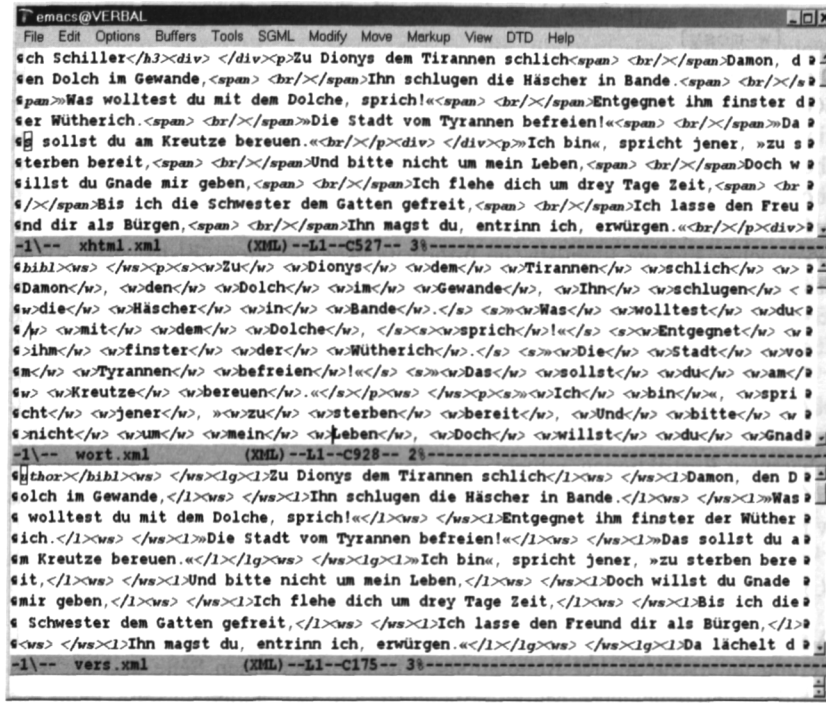

Abbildung 1: Drei separate Annotationen des Beginns der Ballade „Die Bürgschaft"

Die Ballade „Die Bürgschaft" wurde gemäß verschiedener Ebenen annotiert. Eine naheliegende Aufbereitung der Ballade besteht in der Annotation als Gedicht. Hierfür kann das sogenannte TEI Base-Tag-Set ‚verse' Verwendung finden (Sperberg-McQueen/Burnard 1994). Eine weitere Aufbereitung der Ballade erfolgte gemäß XHTML. Des Weiteren wurde der Text nach verschiedenen linguistischen Kriterien annotiert.[19] In der Abbildung sind drei Annotationen wiedergegeben. Die Annotationen sind in separaten Dateien abgespeichert und nicht durch Querverweise miteinander verknüpft. Bei der Darstellung wird der gesamte Dateninhalt in einer Zeile, d.h. ohne die üblichen, der Lesbarkeit dienenden Einrückungen im Editor angezeigt. Dies soll verdeutlichen, dass der Text als eine Kette aufeinanderfolgender Zeichen aufgefasst wird, die in jeder

et al. (2002) verwenden die in Fußnote 17 erwähnte Technik des Stand-Off Markup und speichern ihre reich annotierten Korpora in einer relationalen Datenbank.

19 Die entsprechenden Dateien werden auf der WWW-Seite zum Buch bereitgestellt.

der Annotationen exakt identisch ist, d.h., dass sich die Dateien nur im Markup unterscheiden.

Im nächsten Abschnitt soll gezeigt werden, dass es möglich ist, derartige separate XML-Dokumente für weitere Verarbeitungen zu nutzen. Dies wird exemplarisch an einer Prolog-basierten Repräsentation für annotierte Texte ausgeführt; es wäre allerdings ohne Weiteres möglich statt Prolog andere Formalismen, wie z.B. XPath, als Basis für eine implementierungsnahe Repräsentation der separaten XML-Dokumente zu verwenden. Die der Repräsentation zugrunde liegende Idee besteht darin, auf komplexe Verknüpfungsverfahren zu verzichten, wenn die Datenhaltung in der beschriebenen Form erfolgt, da die annotierten Daten selbst als Grundlage der Verknüpfung dienen können.

4.4 Prolog-Repräsentation von XML-Instanzen

Sperberg-McQueen *et al.* (2001, 2002) diskutieren Möglichkeiten der Interpretation der Bedeutung annotierter Dokumente. Zur Explizierung ihres Ansatzes werden die ausgezeichneten Dokumente in der logikbasierten Programmiersprache Prolog repräsentiert, d.h. jedes Element, jedes Attribut und der Textinhalt wird als Prolog-Fakt gespeichert.[20] Dieser Ansatz ist derart erweitert worden, dass auch die im vorangehenden Abschnitt beschriebenen multiplen Annotationen repräsentiert werden können (vgl. Witt 2002). Hierdurch können alle separaten Annotationen in einer Datenbasis vereinigt werden, die dann z.B. zur automatischen Ermittlung der in Abschnitt 3 thematisierten Beziehungen der Annotationsebenen verwendet werden kann. Hierfür wird – in der einfachsten Fassung – für jedes Element, jedes Attribut und jeden Textknoten jeder Annotationsebene ein Prolog-Fakt erstellt, in dem folgende Informationen enthalten sind:

1. Ein Verweis auf die Annotationsebene;
2. Angaben über die absolute Startposition, der durch das Markup betroffenen Textstellen;
3. Angaben über die Endposition;
4. die Position der betroffenen Einheit in der Baumrepräsentation der Annotationsebene; sowie
5. der Name des Elements (bzw. des Attributs).

Als Beispiel sollen einige Fakten aus der Prolog Repräsentation der verschiedenen Versionen der „Bürgschaft" dienen.

---

[20] Einen ähnlichen Ansatz stellt Schröder (1998) vor, dessen System Pro-SGML die Prolog-Faktenbasis als Grundlage für ein strukturgeleitetes Textretrieval verwendet. Sperberg-McQueen *et al.* (1994) verwenden ihre Prolog-Fakten als Basis für die Repräsentation der Semantik von Markup.

```
node('phrase.xml',34,43,[1,1,5,1,1],element('phr')).
node('vers.xml',34,64,[1,1,1,5,1],element('l')).
node('wort.xml',34,36,[1,1,5,1,1],element('w')).
node('xhtml.xml',34,278,[1,2,5],element('p')).
```

Die erste Strophe beginnt mit der Zeile (<l>) „Zu Dionys dem Tirannen schlich". In den multiplen Annotationen beginnt die Zeichenkette an der absoluten Position 34.[21] An dieser Stelle beginnen das Wort „Zu", die Phrase „Zu Dionys", die gesamte Zeile und die in der XHTML als Abschnitt (<p>) annotierte Strophe. Es ist zu sehen, dass die oben aufgelisteten Informationen in gegebener Reihenfolge als Argumente des Faktes ‚node' vorkommen. Eine derartige Faktenbasis enthält alle Informationen der verschiedenen Annotationen und kann als Datengrundlage für sehr verschiedene Prolog-Prädikate dienen.

5 Resümee und Ausblick

Wer Sprachdaten annotiert, wird mit den Grenzen SGML-basierter Auszeichnungssprachen konfrontiert. Dies liegt insbesondere darin begründet, dass linguistische Daten multidimensionaler Natur sind. Es wurden Techniken beschrieben, die es erlauben, innerhalb und außerhalb von XML multidimensionale Modellierungen vorzunehmen. In einem gegenwärtig laufenden Forschungsprojekt findet der Ansatz der multiplen Annotation der Sprachdaten Anwendung (vgl. Sasaki *et al.* 2002). In diesem Projekt wird auch die beschriebene Prolog-Faktenbasis verwendet. Die Ergebnisse dieser Arbeiten werden auf den WWW-Seiten der DFG Forschergruppe „Texttechnologische Informationsmodellierung "[22] dokumentiert.

6 Literatur

Asahara, Masayuki, Ryuichi Yoneda, Akiko Yamashita, Yasuharu Deny und Yuji Matsumoto: Use of XML and Relational Databases for Consistent Development and Maintenance of Lexicons and Annotated Corpora. In: Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002), Las Palmas 2002, S. 1372-1378

Barnard, David T., Lou Burnard, Jean-Pierre Gaspart, Lynne A. Price, C.M. Sperberg-McQueen und Giovanni Battista Varile: Hierarchical Encoding of Text: Technical Problems and SGML Solutions. In: *Ide, Nancy und Jean Véronis* (Hg.): Text Encoding Initiative: Background and Context. Dordrecht: Kluwer 1995, S. 211-231.

Bird, Steven und Mark Liberman: A formal framework for linguistic annotation. In: Speech Communication 33 (1-2), 2001, S. 33-60

Bray, Tim, Dave Hollander und Andrew Layman (Hg.): Namespaces in XML. W3C Recommendation, World Wide Web Consortium 1999.

Durand, David G.: Palimpsest: Change-Oriented Concurrency Control for the Support of Collaborative Applications. Dissertation, Boston University 1999.

---

21 Vor dieser Stelle findet sich noch der Titel und der Name des Autors.

22 http://www.text-technology.de oder http://coli.lili.uni-bielefeld.de

Durusau, Patrick und Matthew Brook O'Donnell: Concurrent Markup for XML Documents. In: Proceedings of XML-Europe, Barcelona 2002. [http://www.idealliance.org/papers/]

Goldfarb, Charles F.: The SGML handbook. Oxford: Clarendon Press 1990.

Heyer, Gerhard und Christian Wolff (Hg.): Linguistik und neue Medien. Wiesbaden: DUV 1998.

Huitfeldt, Claus und C.M. Sperberg-McQueen: TexMECS: An experimental markup meta-language for complex documents. 2001. [http://www.hit.uib.no/claus/mlcd/papers/texmecs.html]

Ide, Nancy und Jean Véronis (Hg.): Text Encoding Initiative: Background and Context. Dordrecht: Kluwer 1995.

Lawler, John und Helen Aristar Dry (Hg.): Using Computers in Linguistics: A Practical Guide. London: Routledge 1998.

Lobin, Henning (Hg.): Text im digitalen Medium. Linguistische Aspekte von Textdesign, Texttechnologie und Hypertext Engineering. Wiesbaden: Westdeutscher Verlag 1999.

Lobin, Henning: Informationsmodellierung in XML und SGML. Berlin, Heidelberg: Springer-Verlag 2000.

McKelvie, David, Amy Isard, Andreas Mengel, Morten Baun Møller, Michael Grosse und Marion Klein: The MATE workbench - An annotation tool for XML coded speech corpora. In: Speech Communication 33 (1-2), 2001, S. 97-112.

Renear, Allen, Elli Mylonas und David Durand: Refining Our Notion of What Text Really Is: The Problem of Overlapping Hierarchies In: International Association for Literary and Linguistic Computing: Selected papers from the ALLC, ACH Conference: Christ Church, Oxford, April 1992. Oxford: Clarendon Press 1996.

Sasaki, Felix, Claudia Wegener, Andreas Witt, Dieter Metzing und Jens Pönninghaus: Co-reference annotation and resources: a multilingual corpus of typologically diverse languages. In: Proceedings of the 3rd International Conference on Language Resources and Evaluation (LREC-2002), Las Palmas 2002, S. 1225-1230.

Schröder, Bernhard: Pro-SGML: Ein Prolog-basiertes System zum Textretrieval. In: *Heyer, Gerhard und Christian Wolff* (Hg., 1998), S. 205-216.

Simons, Gary F.: The Nature of Linguistic Data and the Requirements of a Computing Environment for Linguistic Research. In: *Lawler, John und Helen Aristar Dry* (Hg., 1998), S. 10-25.

Simons, Gary F. und D. Terence Langendoen: Rationale for the TEI Recommendations for Feature-Structure Markup. In: *Ide, Nancy und Jean Véronis* (Hg., 1995), S. 191-210.

Soria, Claudia, Niels Ole Bernsen, Niels Cadée, Jean Carletta, Laila Dybkjær, Stefan Evert, Ulrich Heid, Amy Isard, Mykola Kolodnytsky, Christoph Lauer, Wolfgang Lezius, Lucas Noldus, Vito Pirrelli, Norbert Reithinger und Andreas Vögele: Advanced Tools for the Study of Natural Interactivity. In: Proceedings of LREC 2002, S. 357-363.

Sperberg-McQueen, C.M. und Lou Burnard (Hg.): Guidelines for Electronic Text Encoding and Interchange (TEI P3). Chicago und Oxford: Text Encoding Initiative 1994.

Sperberg-McQueen, C.M. und Huitfeldt, Claus: Concurrent Document Hierarchies in MECS and SGML. In: Literary and Linguistic Computing 14 (1), 1999, S. 29-42.

Sperberg-McQueen, C.M., Claus Huitfeldt und Allen Renear: Meaning and interpretation of markup. In: Markup Languages: Theory & Practice 2 (3), 2001, S. 215-234.

Sperberg-McQueen, C.M., David Dubin, Claus Huitfeldt und Allen Renear: Drawing inferences on the basis of markup. In: Proceedings of XML-Europe, Barcelona 2002.

Tennison, Jeni und Wendell Piez: The Layered Markup and Annotation Language. Presented at Extreme Markup 2002. [http://xml.lmnl.org/]

Thompson, Henry S. und David McKelvie: Hyperlink Semantics for Standoff Markup of Read-Only Documents. In: SGML '97 Conference Proceedings, Barcelona 1997, S. 227-229.

Witt, Andreas: SGML und Linguistik. In: *Lobin, Henning* (Hg., 1999), S. 121-153

Witt, Andreas: Multiple Informationsstrukturierung mit Auszeichnungssprachen. XML-basierte Methoden und deren Nutzen für die Sprachtechnologie. Dissertation, Universität Bielefeld 2002. [http://archiv.ub.uni-bielefeld.de/disshabi/2002/0007.pdf]