

# **WELSCHER GAST DIGITAL**

## **TEI-HANDBUCH**

*Jakub Šimek*

Version: 0.6  
15. 1. 2014

## Vorwort

§1 Das Projekt ›Welscher Gast digital‹ ist ein Kooperationsprojekt des Sonderforschungsbereichs 933 ›Materiale Textkulturen‹ (Teilprojekt B06) an der Universität Heidelberg und der Universitätsbibliothek Heidelberg. Das Ziel des Projekts ist eine umfassende und innovative digitale Edition des ›Welschen Gastes‹ Thomasins von Zerklare. Die vorliegenden Versuche, eine Konvention für die Kodierung der Projektdaten nach dem TEI-XML-Standard (P5) festzulegen, sind von der Bemühung geprägt, teilweise als Modell für andere ähnliche Editionsprojekte dienen zu können.

Projekt &amp; Ziele

§2 Die Kodierungskonvention von ›Welscher Gast digital‹ beschränkt sich von vornherein auf keine TEI-Customisierung, sondern arbeitet potenziell mit dem gesamten Standard. Zur internen Qualitätskontrolle werden jedoch eigene spezifische Schematron-Regeln eingesetzt.

Customisierung

§3 Es wird zwischen zwei Kodierungsperspektiven für die Erfassung des Werkes und seiner Handschriften unterschieden: werkbasiert und codexbasiert. Die werkbasierte Perspektive strukturiert die Daten primär nach hierarchischen Einheiten des abstrakt gedachten Werkes: Büchern, Kapiteln, Verspaaren, Versen, Wörtern etc. Die codexbasierte Perspektive legt hingegen jeweils eine physische Handschrift zugrunde. Praktisch erfolgt die Transkriptionsarbeit des Projekts in werkbasierter Perspektive, während in codexbasierter Perspektive genuin lediglich die physische Lagenstruktur der Codices erfasst wird. Transformationskripte können jedoch die codexbasierte Erfassung eines Codex um sämtliche in werkbasierter Perspektive für den jeweiligen Codex kodierten Daten ergänzen, sodass eine Datei im Wesentlichen alle Daten eines Codex umfasst und als Grundlage eines digitalen Faksimiles mit eingebetteter Transkription dienen kann.

Kodierungs-  
perspektiven:  
werkbasiert &  
codexbasiert

§4 Die Eingabe und Speicherung der textuellen Daten (v.a. Transkriptionen) erfolgt in UTF-8. Alle Sonderzeichen sind direkt eingefügt (nicht als Zahlencodes oder Entitäten). Deshalb ist für die richtige Darstellung eine Schriftart mit entsprechender Unterstützung nötig. Die Verwendung einiger weniger Zeichen aus der PUA des Unicode-Standards nach den Empfehlungen der Medieval Unicode Font Initiative (MUFI; s. Dokumentation der verwendeten Sonderzeichen) macht den Einsatz einer MUFI-konformen Schriftart empfehlenswert, z.B. Junicode. Für die richtige Darstellung der Sonderzeichen im Web sollte die Schriftart in die jeweilige Webseite eingebettet werden. Unter Umständen kann für bestimmte Darstellungsweisen im Web und für Suchzwecke eine Vereinheitlichung von Buchstabenformen (Allographe von *d*, *r*, *s* etc.) sinnvoll sein.

Zeichenkodierung

§5 Textknoten können am Anfang und am Ende Leerzeichen und Zeilenumbrüche enthalten, die nicht zum Inhalt gehören und bei Transformationen getilgt werden sollten.

Whitespaces  
in Textknoten

## Werkbasierte Perspektive

§6 Die Basisdaten des Werkes sind auf zwölf Dateien im Verzeichnis

Dateien

Textdaten verteilt: Textdaten\_Buch\_00.xml usw. bzw. Textdaten\_Prosa-vorrede.xml. Außerdem enthält die Datei Character\_Declarations.xml die Deklarationen spezieller Glyphen.

§7 Das Werk gliedert sich in Bücher, Kapitel, Verspaare und Verse. In der jeweiligen TEI-Datei ist dieser Inhalt in `text/body` enthalten. Die oberste Ebene ist dabei `<div type="book">`. Diese `<div>`-Elemente tragen jeweils eine `@xml:id` in Form von `b_0`, `b_1` usw. Unterhalb dieser Ebene folgen die Überschriften als `<head>` sowie die Kapitel als `<div type="chapter">`. Deren `@xml:id` hat die Form `ch_5_3`, hier beispielhaft für Kapitel drei in Buch fünf (außerdem gibt es in der Regel am Ende des jeweiligen Buches die Übergangskapitel, z.B. `<div type="chapter" xml:id="ch_1_transition">` am Ende von Buch 1. Unterhalb der Kapitelebene folgen Verspaare, z.B. als `<lg xml:id="vp_141">`, wobei die Nummer jeweils der Nummer des ersten Verses im Verspaar entspricht, hier z.B. `<l xml:id="v_141">`. Die Gliederung in Verspaare dient v.a. der Reimanalyse. Weitere mögliche Kindelemente von `<div type="chapter">` sind `<pb>`, `<cb>` sowie `<fw>`. Lediglich am Anfang von `ch_0_0` steht als direktes Kindelement `<l xml:id="v_0">` die Überschrift des Werkes.

Gliederung

§8 Aus Gründen der reibungslosen Transformation der Gliederungseinheiten in die codexbasierte Perspektive sind Bücher und Kapitel alternativ mit leeren `<milestone>`-Elementen (jeweils `@unit="book"` bzw. `"chapter"`) als Kindelemente der genannten `<div>`-Container kodiert. Das jeweilige Attribut `@n` entspricht der Nummer nach `b_` bzw. `ch_` in der `@xml:id` der übergeordneten `<div>`-Container. Es ist darauf zu achten, dass die genannten Marker beim Zusammentreffen mit Seiten- bzw. Spaltenumbrüchen jeweils *nach* diesen gesetzt werden, damit in der codexbasierten Perspektive hierarchische Einheiten ggf. am Anfang einer Seite oder Spalte anfangen (und nicht am Ende der vorausgehenden Seite oder Spalte).

&lt;milestone&gt;

§9 Neben dem eigentlichen Textinhalt in `<lg>` bzw. `<l>` enthalten jeweilige Kapitel-Container strukturelle Informationen zu einzelnen Handschriften: `<pb>`, `<cb>` und `<fw>`. Diese stehen als Kindelemente von `<div type="chapter">` zwischen den `<lg>`-Elementen. Ist ein Umbruch oder eine andere strukturelle Angabe innerhalb eines Verspaares nötig, erscheinen diese Elemente auch zwischen `<l>`-Elementen, also als Kindelemente von `<lg>`.

Handschriftenorientierte Gliederung

§10 Seitenumbrüche in den Handschriften werden mit `<pb>` markiert. Das Attribut `@edRef` verweist dabei auf die Handschrift, `@fac` auf die Seite, die mit `<surface type="page">` in der codexbasierten Perspektive definiert ist, z.B. `<pb edRef="#A" facs="#A_001r"/>`. Das Element `<cb>` dient der Kodierung des Spaltenumbruchs. Die Spaltenmarkierung erfolgt aus Gründen der Datenkonsistenz auch beim einspaltigen Layout. Beispiele:

Seitenumbrüche und Textblockumbrüche



Beginn der Seite 2r in der Handschrift A:

```
<pb edRef="#A" facs="#A_002r"/>
```

```
<cb edRef="#A" facs="#A_002r-a"/>
```



Beginn der Seite 12v in der Handschrift G:

```
<pb edRef="#G" facs="#G_012v"/>
```

```
<cb edRef="#G" facs="#G_012v-a"/>
```

rechte Spalte:

```
<cb edRef="#G" facs="#G_012v-b"/>
```

§ 11 Das Element `<fw>` dient der Kodierung von mittelalterlichen Seitenzahlen, Seitenüberschriften, Kolumnenüberschriften, Lagenzahlen, Kolophonen usw., die außerhalb des eigentlichen Textkörpers stehen. Das erste Attribut `@corresp` enthält die Zuordnung zur Handschrift. Das zweite Attribut `@type` beschreibt die Kategorie des Phänomens. Bisher kommen in den Daten folgende Typen vor: `<fw type="page_header">` für Seitenüberschriften, `<fw type="quire_numeral">` für Lagenzahlen sowie `<fw type="catchword">` für Wortreklamanten. Des Weiteren tragen `<fw>`-Elemente `@facs`-Attribute, die auf `<zone>`-Elemente in codexbasierter Perspektive verweisen (bzw. die in codexbasierter Perspektive generiert werden, künftig unter Einschluss von Koordinatendaten aus MTK-Online). In `@place`-Attributen ist ferner eine grobe Position auf der Seite angegeben (`top`, `bottom` usw.). Der Inhalt der `<fw>`-Elemente kann komplex sein, also z.B. lemmatisierte `<w>`-Elemente oder aufgelöste Abkürzungen enthalten.

Seitenüberschriften,  
Lagenzahlen u.Ä.

§ 12 `<1>` steht für einen idealen Vers als Gliederungseinheit des Werkes. Dessen Kindelement `<app>` enthält Lesarten einzelner Handschriften für die jeweilige abstrakte Verseinheit. Unterhalb der `<app>`-Ebene befinden sich in drei `<rdgGrp>`-Elementen drei Gruppen von Handschriften bzw. Textzeugen: `<rdgGrp type="gA">` (Handschriften A, G, D, F, War, Si, Bue), `<rdgGrp type="gS">` (Handschriften c, lca, U, W, H, K, M, N, b, E, S; weitere kommen hinzu) sowie `<rdgGrp type="edd">` (Editionen, insbesondere Rue, später auch neuer kritisch hergestellter Text). Innerhalb der `<rdgGrp>`-Elemente folgen mit `<rdg>` die Lesarten einzelner Handschriften.

ideale Verse

§ 13 Da in allen Handschriften des ›Welschen Gastes‹ Verse abgesetzt sind (jeder Vers steht bis auf wenige Ausnahmen auf einer eigenen Zeile), entspricht ein `<rdg>`-Element meist einer Zeile. Deshalb impliziert der Anfang eines `<rdg>`-Elements einen Zeilenumbruch. Wenn innerhalb eines Verses dennoch ein Zeilenumbruch vorkommt, wird er mit `<lb type="in_verse">` markiert. Die `@xml:id` des `<rdg>`-Elements dient zur Identifizierung der physischen Zeile, auf der sich der jeweilige Vers befindet. Sie sieht beispielsweise so aus: `G_009r-a.v_72_line_13`. Dabei steht die letzte Nummer für die laufende Nummer der physischen Zeilen in der Textspalte. Die vorletzte Nummer steht für die Versnummer. Eine gleich strukturierte `@xml:id` erhalten auch Elemente des Typs `<lb type="in_verse">`. Wenn ein Vers nicht auf einer neuen Zeile beginnt, erhält das `<rdg>`-Element eine `@xml:id`, deren Zeilenindikator die tatsächliche Zeilennummer (d.h. dieselbe wie das vorausgehende Element) enthält und auf `cont` endet. Das `<rdg>`-Element trägt außerdem ein Attribut `@prev`, das auf die `@xml:id` des Elements verweist, auf dessen Zeile dieser Vers beginnt (`<rdg>` oder `<lb>`). Dieses Element wiederum, dessen Zeile auch den Beginn des nächsten Ver-

<rdg> versus Zeilen

ses beinhaltet, verweist mit `@next` auf die `@xml:id` des nächsten Verses (deren Zeilenbezeichner auf `cont` endet). Wenn auf einen Zeilenumbruch mitten im Vers eine Zeile folgt, in der dieser selbe Vers immer noch nicht endet (wenn sich also ein Vers über mehr als zwei Zeilen erstreckt), trägt dieses `<lb>`-Element auch das Attribut `@subtype="noendinsight"`. Dessen Attribut `@next` verweist auf die `@xml:id` des nächsten Zeilenumbruchs (dessen `@prev` seinerseits auf die `@xml:id` des „noendinsight“-Zeilenumbruchs verweist). Der Versnummerindikator in der `@xml:id` eines Zeilenumbruchs trägt immer nur die Nummer des Verses, mit dem die Zeile beginnt. Wenn ein Vers nicht auf eine Zeile passt und *über* der Zeile fortgesetzt wird (damit der nächste Vers ohne Unterbrechung auf seiner regulären Zeile anfangen kann), wird ein solcher Zeilenumbruch mit `<lb type="above"/>` gekennzeichnet. Er verweist mit `@prev` auf die `@xml:id` des vorausgehenden `<rdg>`-Elements, das seinerseits mit `@next` auf die `@xml:id` des Zeilenumbruchs verweist. Der Zeilenindikator der `@xml:id` eines solchen Zeilenumbruchs erhält zusätzlich zur Nummer der letzten regulären Zeile einen Buchstaben.

§ 14 Verse, die in der zugrundeliegenden Rückert-Ausgabe nicht vorkommen, aber in den neuen edierten Text aufgenommen werden sollen, werden mit neuen `<l>`- bzw. `<lg>`-Elementen, deren Nummer zusätzlich zur letzten Rückert-Versnummer mit Buchstaben versehen wird, eingefügt. Zusätzliche Verse in einzelnen Handschriften, die nicht in den neuen edierten Text aufgenommen werden sollen, werden (wenn es sich gleichzeitig um eine neue Zeile handelt) mit `<lb type="verse"/>` (oder, wenn gleichzeitig ein Verspaar beginnt: `<lb type="couplet"/>`) markiert und fortlaufend zusammen mit dem letzten regulären Vers im selben `<rdg>` transkribiert. Das `<lb>`-Element erhält eine `@xml:id`, deren Versbezeichner zusätzlich mit einem Buchstaben versehen wird und deren Zeilenindikator der tatsächlichen Zeilennummer auf der Seite entspricht. Beispiel:

```
<lb type="verse" xml:id="A_065r-a.v_4144a_line_32"/>
```

Verse, die in einer Handschrift fehlen, obwohl keine physische Lücke oder Beschädigung vorliegt, werden im entsprechenden `<rdg>`-Element mit `<note>fehlt</note>` markiert. Solche `<rdg>`-Elemente tragen keine `@xml:id` und werden bei der Transformation in die codexbasierte Perspektive nicht übernommen.

§ 15 Wenn es wünschenswert ist, den Einzug einer Zeile in der Kodierung festzuhalten, kann dies mit dem Attribut `@rend="indent"` beim entsprechenden Element `<rdg>` oder `<lb>` vermerkt werden.

§ 16 Vertikale Freiräume zwischen Versen werden mit `<space dim="vertical" extent=... unit=... >` außerhalb von `<lg>` bzw. `<l>`-Elementen notiert. Die Zugehörigkeit zu einer Handschrift wird im Attribut `@corresp`, die Verknüpfung mit einem Koordinatenbereich im Attribut `@facs` angegeben.

§ 17 Die Position von Abbildungen im Text wird mit dem Element `<figure>` angegeben, das außerhalb von `<lg>` bzw. `<l>` steht. Dessen Attribut `@place` gibt die ungefähre Position auf der Seite an, `@facs` verweist auf einen

zusätzliche und  
fehlende Verse

Zeileneinzug

vertikale Freiräume

Abbildungen

Koordinatenbereich. Wenn die Abbildung in einer eindeutigen Relation zur Textstruktur (direkt zwischen Versen) steht, ergibt sich die genaue Position implizit aus der Position des `<figure>`-Elements. Wenn die Relation zum Text nicht eindeutig ist (Position am Rand), wird das Element `<figure>` dorthin gesetzt, wo die engste physische Beziehung zum Text erkennbar ist. Im Kindelement `<span>` und dessen Attributen `@from` und `@to` wird dann der Textbereich (mit `@xml:id` der `<rdg>`- bzw. `<lb>`-Elemente) angegeben, neben dem sich die Abbildung befindet. Die ideelle Beziehung zum Text wird *hier* nicht angegeben.

Mögliche Werte von `@place` bei `<figure>` sind:

<code>inspace</code>	im Textblock
<code>rmargin</code>	am rechten Rand
<code>lmargin</code>	am linken Rand
<code>top</code>	im Seitenkopf
<code>bottom</code>	im Seitenfuß

## Tokenisierung und Lemmatisierung

§ 18 Verse werden zunächst als zusammenhängende Zeichenketten transkribiert. Da das Projekt allerdings eine linguistische Annotation (vor allem Lemmatisierung) anstrebt, ist eine Aufteilung in Wörter notwendig. Jedes Wort wird in ein Element `<w>` gesetzt. Die `@xml:id` der `<w>`-Elemente enthält die Sigle der Handschrift, die Versnummer sowie eine laufende Nummer innerhalb des jeweiligen Verses. Des Weiteren wird das Attribut `@lemmaRef` gesetzt, das die jeweilige Lemma-ID des neuen „Mittelhochdeutschen Wörterbuchs“ enthalten wird. Beispiel:

```
<w xml:id="w_A_142_1" lemmaRef="107376000">Man</w>
<w xml:id="w_A_142_2" lemmaRef="154377000">fol</w>
```

Die lemmaorientierten Grenzen der `<w>`-Elemente stimmen häufig nicht mit der Leerzeichensetzung der Handschriften überein. Wenn eine zusammengeschriebene Zeichenkette in mehrere `<w>`-Elemente aufgetrennt wird, wird die Trennstelle mit `<space extent="0"/>` markiert. Bei Transformationen für Darstellungszwecke sind so getrennte Wörter in der Regel wieder zusammenzufügen. Zeichenketten in `<w>`-Elementen können je nach Schreibweise der Handschriften Leerzeichen enthalten.

§ 19 Ist die Sprache eines Wortes nicht Mittelhochdeutsch (*gmh*), wird das Attribut `@xml:lang` gesetzt. Bisher verwendete Werte sind `lat` (Latein) und `ces` (Tschechisch). Sie kommen insbesondere in den Glossen von Cod. Pal. germ. 330 vor.

§ 20 Sog. komplexe Wortformen (Komposita, bei denen zwischen einzelnen Bestandteilen und der Gesamtform unterschieden werden soll; Pronominaladverbien, Verben mit trennbarem Präfix) werden auf zwei Ebenen lemmatisiert. Im Attribut `@lemmaRef` sind beide Ebenen durch einen Unterstrich getrennt. Vor dem Unterstrich steht die Lemma-ID der Ebene 1 (Ebene

`<w>`  
`<space extent="0"/>`

Sprache

Lemmatisierung  
komplexer  
Wortformen

der Einzelformen), nach dem Unterstrich die Lemma-ID der Ebene 2 (Ebene der Gesamtformen). Die Zusammengehörigkeit einzelner Bestandteile von komplexen Formen wird durch die Verwendung der gleichen fortlaufenden Wortnummer in der @xml:id und eines nach einem Unterstrich folgenden Buchstabens kenntlich gemacht, sofern sich beide Formen innerhalb eines Verses befinden. Beispiel:

```
<rdg wit="#Rue">
<w xml:id="w_Rue_39_1" lemmaRef="">daz</w>
<w xml:id="w_Rue_39_2" lemmaRef="">en</w>
<space extent="0"></space>
<w xml:id="w_Rue_39_3" lemmaRef="">sprich</w>
<w xml:id="w_Rue_39_4" lemmaRef="">ich</w>
<w xml:id="w_Rue_39_5_a" lemmaRef="28932000_29439000">dâ</w>
<w xml:id="w_Rue_39_5_b" lemmaRef="213186000_29439000">von</w>
<w xml:id="w_Rue_39_6" lemmaRef="">n<rhyme n="1">iht</rhyme></w>
</rdg>
```

Wenn auf Ebene 2 zusammengehörige Wortformen nicht im selben Vers stehen, erhalten sie eine normale @xml:id. Der erste Bestandteil verweist dann mit dem Attribut @next auf die @xml:id des später vorkommenden Bestandteils. Dessen Attribut @prev verweist wiederum auf die @xml:id des vorangehenden Bestandteils. Beispiel:

```
<rdg wit="#Rue">
<w xml:id="w_Rue_474_1" lemmaRef="" ana="0_0">ein</w>
<w xml:id="w_Rue_474_2" lemmaRef="" ana="0_0">iegeſlich</w>
<w xml:id="w_Rue_474_3" lemmaRef="" ana="0_0">biderb</w>
<w xml:id="w_Rue_474_4" lemmaRef="" ana="0_0">wirt</w>
<w xml:id="w_Rue_474_5" lemmaRef="" ana="0_0">der</w>
<w xml:id="w_Rue_474_6" lemmaRef="173655000_222399000" next-
="#w_Rue_475_1">t<rhyme n="1">uo</rhyme></w>
</rdg>
```

```
<rdg wit="#Rue">
<w xml:id="w_Rue_475_1" lemmaRef="222069000_222399000" prev-
="#w_Rue_474_6">war</w>
<w xml:id="w_Rue_475_2" lemmaRef="" ana="0_0">ob</w>
<w xml:id="w_Rue_475_3" lemmaRef="" ana="0_0">si</w>
<w xml:id="w_Rue_475_4" lemmaRef="" ana="0_0">alle</w>
<w xml:id="w_Rue_475_5" lemmaRef="" ana="0_0">habent</w>
<w xml:id="w_Rue_475_6" lemmaRef="" ana="0_0">gen<rhyme
n="1">uoc</rhyme></w>
<pc>.</pc>
```

</rdg>

## Abkürzungen

§21 Abgekürzte Wörter werden in abgekürzter und ausgeschriebener Form kodiert. Innerhalb des Elements <w> steht das Element <choice>. Dort befindet sich in <abbr> der abgekürzte Wortlaut des Wortes. Danach folgt das Element <expn> mit dem ausgeschriebenen Wortlaut des Wortes. Der jeweils abgekürzte bzw. aufgelöste Teil des Wortes steht wiederum in den Elementen <am> bzw. <ex>. Sind mehrere Varianten der Auflösung möglich, werden mehrere <expn>-Elemente gesetzt. Wird die Abkürzung durch Superskripte realisiert, wird auch der Basisbuchstabe als Teil der Abkürzung betrachtet. Wenn sich das abgekürzte Wort im Reim befindet, wird der Reim in der ausgeschriebenen Variante markiert.

Abkürzungen

## Reime

§22 Reime, d.h. Teile der jeweils letzten Wörter im Vers ab dem letzten betonten Vokal, werden mit dem Element <rhyme> markiert. Das Attribut @n hält die Anzahl der Reimsilben fest. Bei abgekürzten Wörtern steht die Reimmarkierung nur in der Auflösung der Abbrivatur.

Reime

## Änderungen

§23 Der handschriftliche Text enthält diverse Änderungen, die entweder durch den Schreiber selbst oder durch spätere Personen vorgenommen worden sind. Häufig ist der Zeitpunkt der Änderung nicht feststellbar; insbesondere ist schwer zu erkennen, ob der Schreiber etwas sofort oder erst später korrigiert hat. Deswegen wird der Zeitpunkt einer Änderung nur dann besonders vermerkt, wenn die Änderung erkennbar jüngerem Datums ist (@hand="later"). Bei der Kodierung von Änderungen gilt einerseits die Maxime, dass anhand der Kodierung eine visuell ähnliche Darstellung möglich sein sollte, andererseits der Grundsatz, dass Wörter bzw. einzelne Bearbeitungsphasen von Wörtern durch die Volltextsuche erfassbar sein sollten (d.h. beim Wegfallen von Tags sollte der Text sichtbar sein). Dadurch, dass auf Wortebene gleichzeitig eine linguistische Annotation vorgenommen wird, insbesondere die Markierung von Wörtern durch <w>-Elemente und die Lemmatisierung, und wegen der Markierung der Reime muss ggf. mit komplexen Situationen umgegangen werden. Einige Änderungen sind außerdem in den Handschriften durch Metazeichen gekennzeichnet, die auch entsprechend kodiert werden müssen. Grundsätzlich werden die meisten Änderungsfälle durch <mod>-Elemente markiert. Das Attribut @type legt dabei die Art der Änderung fest. Unterhalb des <mod>-Elements sind dann weitere <mod>-Elemente platziert, einerseits mit einer graphisch handschriftennahen Umsetzung der veränderten Stelle (@type="diplomatic"), andererseits mit einzelnen, durchnummerierten Phasen der veränderten Stelle (@type="phase" @n=""). In <mod @type="diplomatic"> kommen die Elemente <subst>, <del> und <add> vor. Ist in einer Änderungsphase ein Wort mutmaßlich nur unvollständig geschrieben worden (etwa weil der Schreiber den Fehler mitten im Wort gemerkt hat), erhält

Änderungen



das Element `<mod @type="diplomatic">` zusätzlich noch das Attribut `@subtype="incomplete"`.

§24 Einfügungen bzw. Tilgungen eines ganzen Wortes, einer Wortgruppe oder eines Interpunktionszeichens werden durch die Elemente `<add>` bzw. `<del>` ohne den Einsatz von `<mod>` kodiert. Ggf. vorhandene Metazeichen werden als direkte Kindelemente von `<add>` mit dem Element `<metamark>` angegeben. Als Wert des Attributs `@function` kommt bisher `insertion` (zu übersetzen mit „Einfügung“) vor, die Platzierung wird mit `@place` angegeben. Mehrere Metazeichen auf einmal sind möglich.

Einfügungen und  
Tilgungen  
ganzer Wörter

§25 Die Änderungselemente (`<mod>`, `<add>`, `<del>`) und `<w>` werden folgendermaßen kombiniert: Das oberste Element `<mod>` tritt immer als übergeordnetes Element von eventuellen `<w>`-Elementen auf. Die `<w>`-Elemente kommen als Kind-Elemente von `<mod type="phase">` zum Einsatz. Bei alleinstehendem `<add>` bzw. `<del>` (Verwendung ohne `<mod>`) wird `<w>` als Kindelement von `<add>` bzw. `<del>` verwendet.

Kombination  
mit `<w>`

§26 Folgende Änderungstypen werden bei der Verwendung des übergeordneten Elements `<mod>` unterschieden (und im Attribut `@type` vermerkt):

Änderungstypen

<code>addition_in_word</code>	Einfügung im Wort
<code>deletion_in_word</code>	Tilgung im Wort
<code>substitution_in_word</code>	Ersatz im Wort
<code>modification_of_word</code>	Wortänderung
<code>substitution_of_word</code>	Wortersatz
<code>substitution_outside_of_word</code>	Ersatz ohne Wortbezug

§27 Das für Tilgungen eingesetzte Element `<del>` trägt immer das Attribut `@rend`. Dessen Wert beschreibt die Art der Tilgung. Einzelne Werte können miteinander kombiniert werden und sind dann durch Leerzeichen getrennt. Es kommen folgende Werte vor:

`<del>`

<code>strikethrough</code>	durchgestrichen
<code>overwritten</code>	überschrieben
<code>underlined</code>	unterstrichen
<code>overlined</code>	überstrichen
<code>underdotted</code>	unterpungiert
<code>overdotted</code>	überpungiert
<code>erasure</code>	ausradiert
<code>adapted</code>	angepasst (zu einem anderen Buchstaben, muss auch bei <code>&lt;add&gt;</code> vermerkt werden)

§28 Das für Einfügungen und Ergänzungen vorgesehene Element `<add>` trägt in der Regel das Attribut `@place` mit der Angabe der Position der Einfügung. Bisher verwendete Werte sind:

`<add>`

above	über der Zeile
below	unter der Zeile
inline	innerhalb der Zeile
superimposed	darüber (setzt eine Tilgung voraus, in der Regel mit @rend="overwritten")
rmargin	am rechten Rand (der Seite bzw. des Textblocks)
lmargin	am linken Rand (der Seite bzw. des Textblocks)
betweenlines	zwischen den Zeilen (selten für Einfügungen, die sich keiner Zeile zuordnen lassen)

Wenn eine dem Element `<add>` vorausgehende Tilgung das Attribut `@rend="adapted"` trägt, trägt auch das Element `<add>` anstatt des Attributs `@place` dieses Attribut. Diese Kodierung wird verwendet, wenn Buchstaben beim Ersatz nicht getilgt und neu geschrieben, sondern in ihrer Form angepasst wurden.

**§ 29** Einfügung im Wort: Wenn in einem Wort ein oder mehrere Buchstaben eingefügt sind, die die Schreibweise des jeweiligen Wortes korrigieren, aber keine fundamentale Änderung des Wortes bewirken, kommt `<mod type="addition_in_word">` zum Einsatz. Es enthält (wie auch die meisten nachfolgenden Kodierungsvarianten) drei untergeordnete `<mod>`-Elemente: `<mod type="diplomatic">` (Darstellung des Befunds, wie er graphisch in der Handschrift vorliegt), `<mod type="phase" n="1">` und `<mod type="phase" n="2">` (Wort vor der Einfügung und Wort nach der Einfügung). Steht das geänderte Wort im Reim, wird der Reim nach Möglichkeit in allen drei `<mod>`-Elementen markiert. Die Markierung der Einfügung in `<mod type="diplomatic">` erfolgt durch das `<add>`-Element mit der Angabe der Platzierung im Attribut `@place`. Beispiel:

Einfügung  
im Wort

```
<mod type="addition_in_word">
<mod type="diplomatic">n<rhyme n="1"><add place="above">e</add>it</rhyme></mod>
<mod type="phase" n="1">
<w xml:id="w_A_9_7" lemmaRef="">n<rhyme n="1">it</rhyme></w>
</mod>
<mod type="phase" n="2">
<w xml:id="w_A_9_8" lemmaRef="" ana="0_0">n<rhyme n="1">eit</rhyme></w>
</mod>
</mod>
```

**§ 30** Einfügung im Wort mit Metazeichen: Das `<add>`-Element mit dem Eingefügten wird mit einer `@xml:id` versehen. Das die Einfügung signali-

Einfügung  
im Wort  
mit Metazeichen

sierende Metazeichen wird als Kindelement des obersten <mod> kodiert. Mit dem Attribut @target verweist es auf das <add>-Element. Außerdem trägt es das Attribut @function="insertion" sowie das Attribut @place.

```
<mod type="addition_in_word">
<mod type="diplomatic">fiche<add xml:id="seg_w_A_293_4_1" place="above">r</add>leichen</mod>
<mod type="phase" n="1"><w xml:id="w_A_293_4" lemmaRef="">ficheleichen</w></mod>
<mod type="phase" n="2"><w xml:id="w_A_293_5" lemmaRef="">ficherleichen</w></mod>
<metamark function="insertion" target="#seg_w_A_293_4_1" place="above">=</metamark>
</mod>
```

**§ 31** Einfügung eines ganzen Wortes: Das eingefügte Wort ist als Kindelement im Element <add> enthalten.

Einfügung  
eines ganzen Wortes

```
<add place="above"><w xml:id="w_A_606_2" lemmaRef="">f1</w></add>
```

**§ 32** Einfügung eines ganzen Wortes mit Metazeichen:

Einfügung  
eines ganzen Wortes  
mit Metazeichen

```
<add place="above">
<w xml:id="w_A_1046_3" lemmaRef="" ana="0_0">ev</w>
<metamark function="insertion" place="above">^</metamark>
</add>
```

**§ 33** Einfügung eines ganzen Wortes mit zwei Metazeichen, hier auch in Kombination mit Interpunktionszeichen:

Einfügung  
eines ganzen Wortes  
mit zwei  
Metazeichen

```
<add facs="#A_009r_marg" place="rmargin">
<pc>·</pc>
<space extent="0"></space>
<w xml:id="w_A_545_2" lemmaRef="" ana="0_0">iege1ich</w>
<pc>·</pc>
<metamark function="insertion" place="inline"><hi rend="decoration-color:red">ø</hi></metamark>
<metamark function="insertion" place="rmargin"><hi rend="decoration-color:red">ø</hi></metamark>
</add>
```

**§ 34** Einfügung eines Interpunktionszeichens:

Einfügung  
eines Interpunktions-  
zeichens

```
<w xml:id="w_A_398_1" lemmaRef="" ana="0_0">Schimphen</w>
<add place="above"><pc></pc></add>
<w xml:id="w_A_398_2" lemmaRef="" ana="0_0">daz</w>
```

**§ 35** Tilgung im Wort: Wenn durch eine Tilgung das betroffene Wort nicht fundamental verändert wird, wird das Element <mod type="deletion\_in\_word"> verwendet. Beispiel:

Tilgung im Wort

```

<mod type="deletion_in_word">
<mod type="diplomatic">h<rhyme n="2">êv<del rend="underdotted
overdotted">t</del>te</rhyme></mod>
<mod type="phase" n="1"><w xml:id="w_A_964_6"
lemmaRef="">h<rhyme n="2">êvtte</rhyme></w></mod>
<mod type="phase" n="2"><w xml:id="w_A_964_7"
lemmaRef="">h<rhymen="2">êvte</rhyme></w></mod>
</mod>

```

**§ 36** Tilgung eines Wortes: Das getilgte Wort ist mit seinem `<w>`-Element im Elternelement `<del>` eingeschlossen. Beispiel:

Tilgung eines Wortes

```

<del rend="underdotted overdotted strikethrough">
  <w xml:id="w_A_1631_3" lemmaRef="">niht</w>
</del>

```

**§ 37** Tilgung außerhalb eines Wortes: Das Element `<del>`, das als Kindelement von `<rdg>` auftritt, muss nicht unbedingt ein Element `<w>` oder `<pc>` enthalten, wenn es sich um eine Tilgung von Einzelbuchstaben handelt, die nicht lemmatisiert werden können. Die getilgten Buchstaben stehen dann direkt im Element `<del>`.

Tilgung außerhalb  
des Wortes

**§ 38** Ersatz im Wort: Wenn durch den Ersatz eines oder mehrerer Buchstaben keine fundamentale Änderung des Wortes zustande kommt, erfolgt der Einsatz von `<mod type="substitution_in_word">`. Im untergeordneten `<mod type="diplomatic">` wird die geänderte Stelle vom Element `<subst>` eingeschlossen, das mit `<del>` bzw. `<add>` die ersetzte bzw. die ersetzende Lesart enthält. Im Attribut `@rend` des Elements `<del>` wird die Art der Tilgung festgehalten, im Attribut `@place` des Elements `<add>` die Position der ersetzenden Lesart. In den Elementen `<mod type="phase">` werden die einzelnen Bearbeitungsphasen des Wortes festgehalten. Beispiel:

Ersatz im Wort

```

<mod type="substitution_in_word">
<mod type="diplomatic">be<subst><del rend="overwritten">
<gap extent="1" unit="chars" reason="illegible"></gap></del>
<add place="superimposed">z</add></subst>zert
</mod>
<mod type="phase" n="1">
<w xml:id="w_A_74_4" lemmaRef="">be<gap extent="1"
unit="chars" reason="illegible"></gap>zert</w>
</mod>
<mod type="phase" n="2">
<w xml:id="w_A_74_5" lemmaRef="" ana="0_0">bezzert</w>
</mod>
</mod>

```

Die Elemente `<subst>`, `<add>` und `<del>` in `<mod type="diplomatic">` können auch tiefer geschachtelt sein:

```

<mod type="substitution_in_word">

```

```

<mod type="diplomatic">
<del rend="underdotted overdotted">ver<subst>
<del rend="adapted">z</del><add rend="adapted">l</add>
</subst>at</del>
<add place="inline">verlat</add>
</mod>

<mod type="phase" n="1"><w xml:id="w_A_1006_6" lemmaRef=""
>verzat</w></mod>

<mod type="phase" n="2"><w xml:id="w_A_1006_7" lemmaRef=""
>verlat</w></mod>

</mod>

```

**§39 Wortänderung:** Diese Kodierung (`<mod type="modification_of_word">`) kommt zum Einsatz, wenn ein Wort in seinen Teilen fundamental verändert wird. `<mod type="diplomatic">` kann die Elemente `<subst>`, `<add>` und `<del>` enthalten. Beispiel:

Wortänderung

```

<mod type="modification_of_word">
<mod type="diplomatic">fo<add place="above">l</add></mod>
<mod type="phase" n="1"><w xml:id="w_A_164_2" lemmaRef="">fo</w></mod>
<mod type="phase" n="2"><w xml:id="w_A_164_3" lemmaRef="">fol</w></mod>
</mod>

```

**§40 Wortänderung mit Metazeichen:**

Wortänderung mit Metazeichen

```

<mod type="modification_of_word">
<mod type="diplomatic">
hof<subst><del rend="adapted">h</del>
<add rend="adapted">z</add>
</subst>v<add xml:id="seg_w_A_302_3_1" place="above">
>h</add>t
</mod>
<mod type="phase" n="1"><w xml:id="w_A_302_3" lemmaRef=""
>hofhvt</w></mod>
<mod type="phase" n="2"><w xml:id="w_A_302_4" lemmaRef=""
>hofzvht</w></mod>
<metamark function="insertion" target="#seg_w_A_302_3_1" place="below">-</metamark>
</mod>

```

**§41 Wortersatz liegt vor, wenn ein ganzes Wort durch ein anderes ersetzt wird. Beispiel:**

Wortersatz

```

<mod type="substitution_of_word">
<mod type="diplomatic">
<subst>

```

```

<del rend="underdotted overdotted strikethrough">b<rhyme
n="2">era</rhyme></del>
<add place="inline">g<rhyme n="2">ema</rhyme></add>
</subst>
</mod>
<mod type="phase" n="1"><w xml:id="w_A_800_4"
lemmaRef="">b<rhyme n="2">era</rhyme></w></mod>
<mod type="phase" n="2"><w xml:id="w_A_800_5"
lemmaRef="">g<rhymen="1">a</rhyme></w></mod>
</mod>

```

#### §42 Wortersatz mit (zwei) Metazeichen:

Wortersatz  
mit Metazeichen

```

<mod type="substitution_of_word">
<mod type="diplomatic">
<subst>
<del rend="strikethrough">f</del>
<add place="rmargin">f</add>
</subst>
</mod>
<mod type="phase" n="1"><w xml:id="w_A_2462_3" lemmaRef=""
>f</w></mod>
<mod type="phase" n="2"><w xml:id="w_A_2462_4" lemmaRef=""
>f</w></mod>
<metamark function="insertion" place="above"><g ref="#char_
ins1">δ</g></metamark>
<metamark function="insertion" place="rmargin"><g ref="#char_
ins1">δ</g></metamark>
</mod>

```

§43 Ersatz ohne Wortbezug: Diese Variante kommt dann zum Einsatz, wenn keine Zuordnung zu einem Wort möglich ist, etwa da es sich um etwas Unleserliches oder um Interpunktion handelt. Beispiel:

Ersatz  
ohne Wortbezug

```

<mod type="substitution_outside_of_word">
<subst>
<del><gap extent="3" unit="chars" agent="erasure"></gap></del>
<add><pc>:</pc></add>
</subst>
</mod>

```

§44 Das Element `<subst>` enthält immer genau ein Element `<del>` und ein Element `<add>` als direkte Kindelemente.

`<subst>`

§45 Bei Wortumstellungen innerhalb eines Verses werden einzelne Wörter zunächst in ihrer ursprünglich geschriebenen Reihenfolge transkribiert. Nach dem letzten Wort (bzw. Interpunktionszeichen) folgt dann `<listTranspose>`, dessen Kindelement `<transpose>` mit Pointern (`<ptr`

Wortumstellungen

target="">) auf einzelne Wörter die richtige Reihenfolge explizit angibt. Wenn die Umstellung in der Handschrift durch Metazeichen realisiert ist, werden diese als `<metamark function="transposition">` an der richtigen Stelle im Text gekennzeichnet. Wenn eine Zuordnung einzelner Metazeichen zu konkreten Wörtern möglich ist, wird dies im Attribut `@target` angegeben. Die Platzierung des Metazeichens wird im Attribut `@place` mit den bei `<add>` üblichen Werten vermerkt. Beispiel:

```
<rdg xml:id="A_054r-a.v_3434_line_25" wit="#A"
  facs="#A_054r-a">
  <w xml:id="w_A_3434_1" lemmaRef="">Er</w>
  <w xml:id="w_A_3434_2" lemmaRef="">fant</w>
  <metamark function="transposition" target="#w_A_3434_3"
    place="above">a</metamark>
  <w xml:id="w_A_3434_3" lemmaRef="">ainen</w>
  <w xml:id="w_A_3434_4" lemmaRef="">immer</w>
  <metamark function="transposition" target="#w_A_3434_4"
    place="above">a</metamark>
  <w xml:id="w_A_3434_5" lemmaRef="">chamerere</w>
</pc>
<listTranspose>
  <transpose>
    <ptr target="#w_A_3434_4"></ptr>
    <ptr target="#w_A_3434_3"></ptr>
  </transpose>
</listTranspose>
</rdg>
```

## Lücken, Beschädigungen und Undeutliches

§46 Lücken werden mit `<gap>` gekennzeichnet, beschädigter Text mit `<damage>`, unsicher lesbarer Text mit `<unclear>`. Die Größe einer Lücke wird mit `@extent` und `@unit` angegeben. Innerhalb der Zeile ist bei `@unit` die Maßeinheit `chars` (Buchstaben) üblich. Der Grund für die Lücke wird wahlweise mit den Attributen `@agent` bzw. `@reason` angegeben. Bisher verwendete Werte bei `@agent` sind `erasure` (Rasur), `rubbing` (Abreibung) und `painting` (Übermalung), bei `@reason` `illegible` (unleserlich) sowie `cut_off` (abgeschnitten).

Lücken und  
Beschädigungen

## Editorische Tilgungen und Ergänzungen

§47 Mit editorischen Tilgungen und Ergänzungen wird bei Handschriften-  
transkriptionen sehr sparsam umgegangen. Wenn jedoch etwas nach Meinung des Herausgebers als getilgt anzusehen ist, eine graphische Markierung aber fehlt, kann er solches mit dem Element `<surplus>` kennzeichnen. Hält er den Vorschlag einer Ergänzung für sinnvoll, kann er ihn im Element

Editorische Tilgungen  
und Ergänzungen

`<supplied>` einfügen. Wenn zutreffend, sind die genannten Elemente als Elternelemente von `<w>` bzw. `<pc>` zu verwenden.

## Auszeichnung graphischer Schriftphänomene innerhalb der Zeile

§48 Für die Auszeichnung graphischer Schriftphänomene innerhalb der Zeile kommt das Element `<hi>` zum Einsatz. Dessen Attribut `@rend` spezifiziert die Art der Auszeichnung. Folgende Werte kommen vor:

inzeilige  
Formatierung

<code>deluxe_initial color:xy</code>	Prachtinitialie in überwiegender Farbe xy
<code>initial color:xy</code>	Initiale in Farbe xy
<code>color:xy</code>	Text in Farbe xy
<code>decoration-color:xy</code>	Textdekoration (z.B. Strichverzierung) in Farbe xy
<code>large</code>	größere Schrift
<code>superscript</code>	hochgestellte Schrift

Bei Initialen wird zusätzlich im Attribut `@n` deren Höhe in Zeilen angegeben.

Die `<hi>`-Elemente können mehrfach ineinander verschachtelt sein. Beispiel:

```
<hi rend="color:red">
  <w xml:id="w_A_0_1"><hi rend="large">D</hi>er</w>
</hi>
```

Wenn eine graphische Auszeichnung für mehrere Wörter hintereinander gilt, steht sie als Elternelement der Wörter direkt unterhalb der `<rdg>`-Ebene. Wenn eine graphische Auszeichnung für mehrere Zeilen hintereinander gilt, wird sie in jedem `<rdg>`-Element wiederholt (`<hi>` steht nie oberhalb von `<l>` bzw. `<lg>`).

Bei der Verwendung von `<hi rend="superscript">` ist darauf zu achten, dass das Element `<hi>` nicht innerhalb des Abkürzungsmarkers `<am>` erlaubt ist. Innerhalb von `<am>` sind daher hochgestellte Buchstaben durch entsprechende Unicode-Zeichen zu realisieren.

Überschriften können zusätzlich zur graphisch orientierten Auszeichnung mit `<hi>` außerdem mit dem semantischen Element `<head>` gekennzeichnet werden.

## Besonderheiten bei Initialen

§49 Anweisungen der Schreiber für Initialen werden vor der Initiale selbst und außerhalb des Wortes als Metazeichen kodiert: `<metamark function="cue_initial">`. Die Position der Anweisung wird mit den üblichen Werten im Attribut `@place` vermerkt. Ist die Anweisung unter der ausgeführten Initiale erkennbar, wird in `@place` der Wert `inspace` („an vorgehener Stelle“) angegeben. Nicht ausgeführte Initialen werden innerhalb von `<hi rend="initial">` mit `<space>`, dessen Kindelement `<desc>` und

Besonderheiten  
bei Initialen



einer Erläuterung vermerkt (z.B. S-Initiale nicht ausgeführt). Die Zeilenhöhe des Freiraums wird mit @n bei <hi> angegeben, wie es auch bei ausgeführten Initialen der Fall ist.

## Glossen

§50 Glossen werden jeweils am Ende eines <rdg>-Inhalts mit dem Element <note type="gloss"> vermerkt. Das Attribut @place hält die Position der Glosse fest, das Attribut @target den Bezug auf die @xml:id des entsprechenden Wortes. Wenn sich eine Glossierung auf mehrere Wörter hintereinander bezieht, wird im Attribut @target die @xml:id des ersten und letzten Wortes, durch ein Leerzeichen getrennt, angegeben. Wenn sich eine Glossierung auf mehrere Wörter bezieht, die nicht unmittelbar aufeinander folgen, wird dies mit @subtype="interrupted\_target" vermerkt. Im Attribut @target werden dann einzelne Bezugswörter mit ihren @xml:id-s explizit aufgezählt, jeweils durch Leerzeichen getrennt. Wenn es bei einer Glossierung, die aus mehreren Wörtern besteht, wünschenswert ist, den Bezug einzelner Wörter aufeinander im Einzelnen anzugeben, wird dies beim Element <w> in der Glossierung mit dem Attribut @corresp und der entsprechenden @xml:id aus dem Haupttext vermerkt. Moderne Glossen werden mit dem Attribut @resp="modern" gekennzeichnet. Wenn sich eine Glosse nur auf ein Wortsegment bezieht, wird dieses Segment mit <seg> im Wort markiert und erhält eine @xml:id, z.B. <w><seg xml:id="seg\_w\_A\_2680\_4\_1">1r</seg>s</w>. Die Glosse bezieht sich dann mit @target auf diese @xml:id.

Metazeichen werden bei modernen Glossen nicht vermerkt.

Glossen

## Editorische Anmerkungen

§51 Editorische Anmerkungen werden im Element <note type="editorial"> vermerkt. Der Bezug wird mit dem Attribut @target realisiert. Es darf sich auf Elemente des Typs <w>, <seg>, <hi> und <mod> (oberste Ebene) beziehen. Eine textuelle Hervorhebung in einer Anmerkung, die typischerweise kursiv ausgegeben werden sollte, steht im Element <emph>.

Anmerkungen

## Spezielle Glyphen

§52 Glyphen, die nicht zum Unicode-Standard gehören, sind in der Datei Character\_Declarations.xml verzeichnet. Dazu gehören einerseits Glyphen, die nach den Vorschlägen der MUFI-Initiative in der PUA des Unicode-Standards definiert und in MUFI-kompatiblen Schriftarten enthalten sind, andererseits aber auch Glyphen, die bisher weder standardmäßig definiert noch darstellbar sind. In beiden Fällen wird ein solches Zeichen im Text mit dem Element <g> gekennzeichnet. Dessen Attribut @ref verweist auf die @xml:id des Zeichens in Character\_Declarations.xml. Bei Glyphen aus der PUA wird als Inhalt des <g>-Elements das Zeichen selbst eingesetzt, bei anderen Glyphen ein Ersatzwert des Zeichens. Unvollständig geschriebene Buchstaben werden mit <g type="incomplete"> markiert.

Spezielle Zeichen

## Codexbasierte Perspektive

§53 Die codexbasierte Perspektive wird durch Transformationsskripte aus den werkbasierten Textdaten unter Hinzunahme einer Lagenstruktur-Datei erzeugt. In der Ergebnisdatei befindet sich die gesamte Kodierung einer Handschrift unterhalb des `<teiHeader>` im Element `<sourceDoc>`. Innerhalb dieses Containerelements wird die physische Struktur des Codex durch verschachtelte `<surfaceGrp>`-Elemente genau abgebildet. Das Attribut `@type` spezifiziert dabei die Kategorie der physischen Codexeinheit, jedes Element trägt auch eine `@xml:id`. Es sind folgende Typen vorgesehen:

Lagenstruktur

binding	Einband
gathering	Lage
bifolium	Doppelblatt
leaf	Blatt

Bei Einheiten der Kategorie `binding` und `bifolium` sind deren direkte Kindelemente der Kategorie `leaf` deren physischer Bestandteil. Die Einheiten der Kategorie `leaf` enthalten in der Regel jeweils zwei `<surface>`-Elemente, die das Attribut `@type="page"` tragen und die beiden zum Blatt gehörenden Seiten repräsentieren. Jedes Element `<surface>` trägt eine `@xml:id`, welche zur projektinternen kanonischen Bezeichnung von Seiten dient (z.B. `A_001r`).

Wenn von einem Doppelblatt eine Hälfte fehlt, wird dies so vermerkt:

```
<surfaceGrp type="bifolium" subtype="incomplete_firstleafmissing"> bzw.
```

```
<surfaceGrp type="bifolium" subtype="incomplete_secondleafmissing">
```

Ein zusätzlich eingefügtes Blatt wird an der entsprechenden Stelle mit `<surfaceGrp type="leaf" subtype="added">` vermerkt.

§54 Innerhalb einer Seite werden mit `<zone>`-Elementen Bereiche der Seite angegeben. Typisch ist dabei das Attribut `@type="column"` für Textspalten. `<fw>`-Elemente der werkbasierten Perspektive werden auch in eigene `<zone>`-Elemente gesetzt. Die in der werkbasierten Perspektive in `<rdg>`-Elementen enthaltenen Textinhalte werden Zeile für Zeile in eigene `<line>`-Elemente ausgegeben. Umbrüche von Büchern, Kapiteln, Verspaaren und Versen werden durch `<milestone>`-Elemente innerhalb der `<zone>`-Bereichen realisiert. Das Element `<space dim="vertical">` wird innerhalb von Textspalten platziert.

Seitenstruktur