

TUSTEP

Import & Export

2015

Inhalt

Allgemeines	5
*IMPORT	6
*TAGUM	9
*EXPORT	11
Vorbemerkungen	13
Datenstruktur	14
Zeichenvorrat	15
Silbentrennung	17
Farben	18
Voreinstellungen	19
Absatzgruppen	22
Absätze	25
Listen	32
Trennlinien	34
Zwangsumbruch	35
Zeichenfolgen	36
Fußnoten, Endnoten	42
Anmerkungen	45
Marginalien	47
Daten ignorieren	51
Tags ignorieren	52
Tabulatoren	53
Zentrieren	55
Datum, Uhrzeit, Seite	56
Kopftext, Fußtext	57
Lesezeichen	59
Navigation	59
Verweise	60
Hyperlinks	64
Laufende Nummern	66
Tabellen	68
Verzeichnisse	74
Register	80
Bilder	84
Bastelecke	88
Definitionen	89
Formeln	107
Vordefinierte Tags	120
Stichwörter	140

Allgemeines

Von anderen Programmen beschriebene Textdateien (z. B. XML-Dateien) können mit dem Kommando #UMWANDLE nach TUSTEP importiert werden. Dabei ist der Code anzugeben, in dem die Daten codiert sind; dieser ist meist UTF8 oder ISO:

```
#UMWANDLE, fremddatei, tustepdatei, 0, +, CODE=utf8
```

Mit dem gleichen Kommando können Texte, die in TUSTEP-Dateien gespeichert sind, für andere Programme exportiert werden; auch dabei ist der Code anzugeben, in dem die Daten codiert werden sollen:

```
#UMWANDLE, tustepdatei, fremddatei, 0, +, CODE=utf8
```

Um Daten von Textverarbeitungsprogrammen wie z. B. MS-WORD ohne Formatierung und ohne Auszeichnungen zu übernehmen, können sie mit dem entsprechenden Programm als "Nur Text" in eine TXT-Datei abgespeichert werden; die Daten dieser Datei können dann mit #UMWANDLE nach TUSTEP importiert werden.

Einzelheiten zum Kommando #UMWANDLE sind im TUSTEP-Handbuch beschrieben (→Grundlagen→Kommandos→#UMWANDLE).

Mit TUSCRIPT können XML-Dateien direkt gelesen und auch beschrieben werden, d. h. die Daten müssen vor der Bearbeitung nicht "importiert" und danach wieder "exportiert" werden. TUSCRIPT ist die Script-Sprache von TUSTEP und ist im TUSTEP-Handbuch beschrieben (→Grundlagen→Makros).

Um Daten von Textverarbeitungsprogrammen wie z. B. MS-WORD mit Formatierung und Auszeichnungen zu übernehmen, müssen die Dateien (z. B. die DOC-, DOCX- oder ODT-Dateien) mit dem entsprechenden Programm geladen und als XML-Datei (Word **2003** XML-Dokument, nicht als Word XML-Dokument!) oder als RTF-Datei (Rich-Text-Format) abgespeichert werden; diese Dateien können mit dem Kommando-Makro *IMPORT nach TUSTEP importiert werden. Dabei ist anzugeben, ob bestimmte Informationen zur Formatierung oder Textauszeichnung (z. B. Schriftgröße oder Farbangaben) ignoriert werden können:

```
#*IMPORT, fremddatei, tustepdatei, , +, IGNORIEREN=-
```

Entsprechend können Texte, die in TUSTEP-Dateien gespeichert sind und geeignete Angaben zur Formatierung und Textauszeichnung enthalten, mit dem Kommando-Makro *EXPORT für andere Textverarbeitungsprogramme in RTF-Dateien exportiert werden:

```
#*EXPORT, tustepdatei, fremddatei, , +
```

Wenn beim Exportieren von Daten Tags mit gleichem Namen verschiedene Wirkung haben sollen, je nachdem welchen Tags sie untergeordnet sind, müssen diese gleichnamigen Tags zuvor so umbenannt werden, dass sie einen eindeutigen Namen haben. Tags können mit dem Kommando-Makro *TAGUM umbenannt werden.

Die Kommando-Makros *IMPORT, *EXPORT und *TAGUM sind im Folgenden beschrieben.

*IMPORT

Mit dem Makro *IMPORT können Texte, die in RTF-Dateien oder in XML-Dateien im Format "Word 2003 XML-Dokument" (nicht "Word XML-Dokument!") abgespeichert sind, nach TUSTEP importiert werden.

Das Makro hat folgende Spezifikationen:

QUELLE	= datei	Name der Quelldatei.
ZIEL	= datei	Name der Zieldatei.
MODUS	= -STD-	* (andere Angaben z. Z. noch nicht vorgesehen)
LOESCHEN	= -	* Daten in der ZIEL- und LISTE-Datei nicht löschen.
	= +	Daten in der ZIEL- und LISTE-Datei zuerst löschen.
LISTE	= +	* Liste der Tags ins Ablaufprotokoll ausgeben.
	= -	Keine Liste der Tags ausgeben.
	= datei	Name der Datei für die Liste der Tags.
IGNORIEREN	= -	Alle vorgesehenen Tags und Attribute ausgeben.
	= +	Alle ignorierbaren Tags und Attribute (s. u.) nicht ausgeben.
	= name	Name des zu ignorierenden Tags oder Attributs. Mehrere Namen können durch Apostroph getrennt angegeben werden.
	= !name	Alle ignorierbaren Tags und Attribute (s. u.) mit Ausnahme der angegebenen nicht ausgeben. Mehrere Namen können durch Apostroph getrennt angegeben werden; das Ausrufezeichen ist nur einmal am Anfang anzugeben.

Folgende Namen von Tags bzw. Attributen sind möglich:

head	Vorspann
before	Leerraum vor Absätzen
after	Leerraum nach Absätzen
indent	Einrücken eines Absatzes
align	Ausrichtung eines Absatzes
break	Seitenumbruch
font	Schriftname
size	Schriftgröße
kerning	Kerning
line-feed	Zeilenabstand
scaling	Skalierung
char-space	Zeichenabstand
hyphenation	Silbentrennung ein/aus
tab	Tabulatorspezifikation
bm	Bookmark
hl	Hyperlink

		color	Farben
		navigation	Navigation
ERSETZEN	= -	*	Namen der Formatvorlagen nicht verändern.
	= alt:neu		Alter und neuer Name der Formatvorlage, deren Name geändert werden soll. Mehrere Namenspaare können durch Apostroph getrennt angegeben werden.
SATZLAENGE	= -STD-	*	Datensätze nicht aufteilen.
	= n		Datensätze mit mehr als n Zeichen bei Leerzeichen in mehrere Datensätze mit maximal n Zeichen aufteilen, jedoch möglichst nur bei Leerzeichen außerhalb von Tags (d. h. außerhalb von "<" und ">") aufteilen. Falls zur Spezifikation KENNUNG eine Kennung angegeben ist, dürfen die Datensätze auch unmittelbar vor einem Anfangs-Tag und unmittelbar nach einem Ende-Tag aufgeteilt werden.
	= n-m		Wie n; falls jedoch keine entsprechende Stelle zum Aufteilen des Datensatzes gefunden wird, dürfen die Datensätze bis zu m Zeichen lang werden.
KENNUNG	= -STD-	*	Keine Kennung einfügen.
	= ...		Angegebene Kennung vor den Stellen in die Datensätze einfügen, an denen sie aufgeteilt werden. Als Kennung darf eine beliebige Zeichenfolge angegeben werden; sie darf jedoch keine Leerzeichen, Minuszeichen, Nummernzeichen, Kommata, Gleichheitszeichen, Apostrophe oder Anführungszeichen enthalten.
OPTIONEN	= -	*	Keine Optionen (Normalfall).
	= SYM		Codes F0xx entsprechend dem MS-SYMBOL-Font umcodieren.
	= BLANK		Leerzeichen wie andere Zeichen behandeln.
	= PICTEXT		Auch Text in "Bildern" importieren.
			Es können mehrere Optionen angegeben werden; sie müssen durch Apostroph getrennt sein.
PROTOKOLL	= +	*	Fehlermeldungen ins Ablaufprotokoll ausgeben.
	= datei		Fehlermeldungen in angegebene Datei ausgeben.

Tags, die nicht benötigt werden, sollten nicht erst nachträglich eliminiert werden, sondern zur Spezifikation IGNORIEREN angegeben werden, damit die restlichen Tags optimiert und ggf. reduziert werden können.

Eine Angabe zur Spezifikation ERSETZEN ist notwendig, wenn sich die Namen der Formatvorlagen mit den Namen von vordefinierten Tags überschneiden. Eine Formatvorlage mit dem Namen "i" ergäbe z. B. "<i>"; dieses Tag dient zur Kennzeichnung kursiver Textteile. Damit "<i>" eindeutig bleibt, muss in diesem Fall angegeben werden, welcher Name für die Formatvorlage an Stelle von "i" eingesetzt werden soll.

Beim Importieren werden nicht alle in der QUELL-Datei enthaltenen Informationen berücksichtigt, auch nicht alle, die ggf. bei einem früheren Export der Daten mit dem Makro *EXPORT berücksichtigt wurden.

*TAGUM

Mit dem Makro *TAGUM können Tags umbenannt werden.

Das Makro hat folgende Spezifikationen:

QUELLE	= <code>datei</code>	Name der Quelldatei.
ZIEL	= <code>datei</code>	Name der Zieldatei.
MODUS	= <code>-STD-</code>	* (andere Angaben z. Z. noch nicht vorgesehen)
LOESCHEN	= <code>-</code>	* Daten in der ZIEL-Datei nicht löschen.
	= <code>+</code>	Daten in der ZIEL-Datei zuerst löschen.
TAGS	= <code>datei</code>	Name der Datei mit den Angaben zum Umbenennen.
	= <code>*</code>	Angaben zum Umbenennen folgen auf den Makroaufruf und sind mit <code>*eof</code> abgeschlossen.
TILGEN	= <code>-</code>	* Keine Attribute tilgen.
	= <code>+</code>	Attribute, die eine Attribut-Bedingung erfüllen tilgen.

Jede Angabe zum Umbenennen eines Tags muss in einer eigenen Zeile stehen. Eine solche Angabe besteht aus einem Pfad für ein Tag, das umbenannt werden soll, einem Namen, den das Tag erhalten soll, und optional eine Attribut-Bedingung. Pfad und Name sowie Name und Attribut-Bedingung müssen jeweils durch mindestens ein Leerzeichen voneinander getrennt sein. Der Name kann in spitze Klammern eingeschlossen werden.

Ein Pfad besteht aus mindestens dem Namen des Tags, das umbenannt werden soll; er muss in spitzen Klammern eingeschlossen werden. Damit würden alle Tags mit dem angegebenen Namen umbenannt, die keinem anderen Tag untergeordnet sind. Soll ein Tag auch umbenannt werden, wenn es anderen Tags untergeordnet ist, müssen die übergeordneten Tags davor angegeben werden. Diese Angabe kann sich aus folgenden drei Formen zusammensetzen: `<tag>` für ein Tag mit dem Namen `tag`, `<*>` für ein Tag mit einem beliebigen Namen und `"*"` für null oder beliebig viele Tags mit beliebigem Namen. In Tag-Namen kann `"?"` als Stellvertreter für ein beliebiges Zeichen stehen, jedoch nicht im letzten Tag, dessen Name geändert werden soll.

Eine Attribut-Bedingung hat die Form `attribut=wert1:name1'wert2:name2'...` und bewirkt, dass das Attribut mit dem Namen `attribut` überprüft wird. Hat das Attribut den Wert `wert1`, erhält das Tag den Namen `name1`, hat es des Wert `wert2`, erhält das Tag den Namen `name2` usw.; hat es einen Wert, der in der Attribut-Bedingung nicht aufgeführt ist, erhält es den Namen, der zwischen Pfad und Attribut-Bedingung angegeben ist.

Beispiel: Das Tag <name> soll

- in <autor-name> umbenannt werden, wenn es dem Tag <autor> (direkt oder indirekt) untergeordnet ist,
- in <hrsg-name> umbenannt werden, wenn es dem Tag <herausgeber> (direkt oder indirekt) untergeordnet ist,
- in <pers-name> umbenannt werden, wenn es das Attribut `typ=person` enthält,
- in <orts-name> umbenannt werden, wenn es das Attribut `typ=ort` enthält.

Die Bedingungen werden in der angegebenen Reihenfolge überprüft.

```
#*TAGUM, quelldatei, zieldatei, , +, *
*<autor>*<name>      <autor-name>
*<herausgeber>*<name> <hrsg-name>
*<name>              <name>  typ=person:pers-name'ort:orts-name
*eof
```

Es wird davon ausgegangen, dass die Tags in der QUELL-Datei alle korrekt geschachtelt sind, andernfalls wird das Programm abgebrochen.

*EXPORT

Mit dem Makro *EXPORT können Texte aus TUSTEP-Dateien ins RTF-Format für MS-Word und andere Textverarbeitungsprogramme exportiert werden.

Das Makro hat folgende Spezifikationen:

QUELLE	= datei	Name der Quelldatei.
ZIEL	= datei	Name der Zieldatei.
MODUS	= -STD-	* (andere Angaben z. Z. noch nicht vorgesehen)
LOESCHEN	= -	* Daten in der ZIEL-Datei nicht löschen.
	= +	Daten in der ZIEL-Datei zuerst löschen.
DEFINITIONEN	= datei	Name der Datei mit den Definitionen.
	= -	* Keine Definitionen in separater Datei.
OPTIONEN	= -	* Keine Optionen (Normalfall).
	= BLANK	Alle Leerzeichen exportieren.
ANZEIGEN	= -	* Ergebnis nicht anzeigen.
	= +	Ergebnis anzeigen.
PROTOKOLL	= +	* Fehlermeldungen ins Ablaufprotokoll ausgeben.
	= datei	Fehlermeldungen in angegebene Datei ausgeben.

Der Text kann u. a. Überschriften, Einschaltungen, Aufzählungen, Nummerierte Listen, Fußnoten, Endnoten, Anmerkungen, Marginalien, Tabellen, Verweise, Hyperlinks und mathematische Formeln enthalten. Innerhalb des Textes kann zwischen ein- und mehrspaltigem Satz sowie zwischen Hoch- und Querformat (z. B. für breite Tabellen) gewechselt werden. Mit Hilfe von Tabellen ist auch mehrspaltiger Parallelsatz möglich. Sind die Einträge im Text entsprechend gekennzeichnet, kann MS-Word Verzeichnisse und Register mit den dann aktuellen Seitennummern automatisch erstellen.

Um aus einer TUSTEP-Datei eine PDF-Datei zu erstellen, kann die Datei mit *EXPORT in eine RTF-Datei exportiert, mit MS-Word geladen und dann als PDF-Datei abgespeichert werden. Dabei werden auch die Einträge der Navigationsleiste als "Lesezeichen", sowie die Verweise und die Hyperlinks mitübertragen, wenn beim Abspeichern die Option "Textmarken erstellen mit Hilfe von *Überschriften*" aktiviert ist.

Die exportierten Daten werden von MS-Word 2010 korrekt interpretiert und dargestellt. Bei anderen Textverarbeitungsprogrammen kann es bei komplexen Formatierungsanforderungen (z. B. bei Marginalien, Formeln oder automatisch erstellten Verzeichnissen) Einschränkungen geben. Zu dieser Problematik schreibt die Zeitschrift *c't* in der Ausgabe 3/2011 auf Seite 106 zu OpenOffice und LibreOffice: „In beiden Paketen arbeitet der Import nur mangelhaft und ist bestenfalls für sehr einfache Textdokumente brauchbar. Selbst in sehr einfach gestalteten Textdokumenten wurde der Text durchweg anders als in Word umbrochen und Bilder fast immer falsch platziert. Zwar übernehmen OpenOffice und LibreOffice Überschriften, Nummerierungen und Fußnoten, formatieren das Dokument aber vollkommen anders als das Original.“

Soll eine mit *EXPORT erstellte Datei an jemand weitergegeben werden, der kein MS-Word hat, kann es sinnvoll sein, die RTF-Datei mit MS-Word zu laden, als DOC-Datei

(nicht als DOCX-Datei und auch nicht als ODT-Datei) abzuspeichern und dann die DOC-Datei weiterzugeben. Die anderen Textverarbeitungsprogramme liefern mit DOC-Dateien oftmals bessere Ergebnisse als mit RTF-Dateien. Soll eine Datei nur zum Lesen bzw. Ausdrucken weitergegeben werden, ist es zweckmäßig, die RTF-Datei mit MS-Word zu laden, als PDF-Datei abzuspeichern und dann die PDF-Datei weiterzugeben.

Um den Text entsprechend auszuzeichnen, können vordefinierte und selbst definierte Tags verwendet werden. Jedes Tag muss vollständig in einem Datensatz stehen, andernfalls wird es nicht als Tag erkannt. Die Definitionen der eigenen Tags müssen entweder am Anfang der QUELL-Datei vor dem Text oder in der zur Spezifikation DEFINITIONEN angegebenen Datei stehen. Welche Tags vordefiniert sind und wie eigene Tags definiert werden können, ist im Folgenden beschrieben.

Vorbemerkungen

Export-Tags

Zum Formatieren des Textes sind Tags vordefiniert. Zusätzlich können eigene Tags definiert, auch vordefinierte umdefiniert werden. Deshalb ist es nicht erforderlich, die eigenen Tags im Text durch vordefinierte Tags zu ersetzen. Alle in dieser Beschreibung aufgeführten Tags werden beim Exportieren in entsprechende RTF-Anweisungen umgesetzt.

Import-Tags

Einige in dieser Beschreibung aufgeführten Tags (z. B. für 17 verschiedene Unterstreichungen) scheinen nicht sehr nützlich zu sein. Sie sind nur der Vollständigkeit halber aufgeführt, damit diese Beschreibung auch zum Nachschlagen der Tags verwendet werden kann, die in importierten Daten vorkommen können.

Attributwerte

Groß- und Kleinschreibung ist bei den Attributwerten für *EXPORT belanglos. Die XML-Konventionen verlangen, dass Attributwerte in Anführungszeichen eingeschlossen werden, die HTML-Konventionen verlangen dies nicht. Von zwei Ausnahmen abgesehen, sind deshalb beide Möglichkeiten erlaubt. Wenn der Attributwert (z. B. ein Font-Name) ein Leerzeichen enthält, muss er in Anführungszeichen eingeschlossen werden, und wenn der letzte Attributwert in einem Anfangs-Tag mit einem Prozentzeichen endet, muss er in Anführungszeichen eingeschlossen werden, weil sonst das Prozentzeichen zusammen mit der nachfolgenden spitzen Klammer, die das Tag abschließt, als Akzent-Codierung interpretiert würde.

Maßeinheit Punkt

Als Maßeinheit für Schriftgröße, Einrückungen, Abstände usw. dient der Punkt (DTP-Punkt-System, nicht Didot-Punkt). 72 Punkt = 1 Zoll = 2,54 cm. Die Angaben sind auf einen zwanzigstel Punkt genau möglich, z. B. 10.5 (mit Dezimalpunkt, nicht mit Dezimalkomma) für zehneinhalb Punkt. An Stelle von Punkt können auch Millimeter (z. B. 15mm), Zentimeter (z. B. 2cm) oder Prozent (z. B. 10%) angegeben werden, wobei sich die Prozentzahl auf die Breite bzw. Höhe des Satzspiegels bezieht. Diese alternativen Angaben werden vom Programm in Punkt umgerechnet und können an jeder Stelle verwendet werden, an der in dieser Beschreibung für einen Attributwert eine Angabe in Punkt (p) verlangt wird. Ausnahme: Die Schriftgröße muss in Punkt auf einen halben Punkt genau angegeben werden.

Beispiele

Beispiele sind in dieser Beschreibung jeweils zwischen zwei durchgehenden Linien eingeschlossen und durch eine kurze Linie unterteilt. Im oberen Teil wird der Text angezeigt, wie er von MS-Word dargestellt wird. Im unteren Teil wird der gleiche Text mit den Tags in Courier-Schrift angezeigt, wie er in der TUSTEP-Datei steht.

Datenstruktur

Wenn die Definitionen der eigenen Tags in der QUELL-Datei stehen, können die Daten in der Quell-Datei folgende Struktur haben:

```
<document>
  <head>
    Definitionen
  </head>
  <body>
    Text
  </body>
</document>
```

Die drei nachfolgend aufgeführten Tags sind alle optional. Wenn sie vorkommen, dürfen sie nur in der oben gezeigten Anordnung vorkommen.

```
<document> Dokument
<head>      Definitionen
<body>     Text
```

Diese drei Tags werden ignoriert; jedes dieser Tags darf beliebige Attribute enthalten.

Wenn die Definitionen der eigenen Tags in einer separaten Datei stehen, sind die drei oben aufgeführten Tags nicht vordefiniert; sie müssen ggf. selbst definiert werden.

Es wird davon ausgegangen, dass jedes Tag in *einer* Zeile steht, dass alle Tags korrekt geschachtelt sind und dass die Syntax der Tags stimmt. Um dies sicherzustellen, sollten die Daten vor dem Exportieren z. B. mit der Anweisung `tp` im TUSTEP-Editor überprüft werden. Das gleiche gilt auch für die Definitionen, wenn sie in einer anderen Datei stehen.

Beispiel für einen Text ohne Tags

Wie dieses Beispiel zeigt, müssen einfach formatierte Texte nicht unbedingt mit Tags versehen sein. Mit einer Leerzeile kann ein neuer Absatz begonnen werden. Jede weitere unmittelbar folgende Leerzeile wird als solche übernommen.

Im Text können Auszeichnungen auch mit konventionellen Codes erreicht werden, z. B. *kursiv* oder KAPITÄLCHEN. Weitere Codes sind im Kapitel "Zeichenfolgen" aufgeführt.

```
#1+Beispiel für einen Text ohne Tags#1-
```

Wie dieses Beispiel zeigt, müssen einfach formatierte Texte nicht unbedingt mit Tags versehen sein. Mit einer Leerzeile kann ein neuer Absatz begonnen werden. Jede weitere unmittelbar folgende Leerzeile wird als solche übernommen.

Im Text können Auszeichnungen auch mit konventionellen Codes erreicht werden, z. B. `#+kursiv#/-` oder `#k+Kapitälchen#k-`. Weitere Codes sind im Kapitel "Zeichenfolgen" aufgeführt.

Zeichenvorrat

Es steht der gesamte TUSTEP-Zeichenvorrat zur Verfügung, soweit in Unicode ein entsprechender Code vorgesehen ist. Darüber hinaus können sämtliche Unicode-Zeichen in der Form # [xxxx] verwendet werden, wobei xxxx der hexadezimale Code des Zeichens ist.

Sinnvollerweise sollten jedoch nur Zeichen verwendet werden, die im jeweils verwendeten Font auch vorhanden sind, da sie sonst nicht dargestellt werden können. Falls ein Font eine andere Codierung als Unicode voraussetzt, müssen die Zeichen, deren Codierung nicht Unicode entspricht, zuvor in die Form # [xx] bzw. # [xxxx] umgewandelt werden, wobei xx bzw. xxxx der hexadezimale Code des Zeichens im jeweiligen Font ist. Hierzu können Code-Tabellen (siehe Seite 105) definiert werden.

Für einige Sonderzeichen gibt es vordefinierte Tags:

<code><enspace/></code>	#[2002]	Spatium, etwa ein halbes Geviert breit
<code><emspace/></code>	#[2003]	Spatium, etwa ein ganzes Geviert breit
<code><hyphen/></code>	-	#[2010] Strich, etwa ein viertel Geviert lang
<code><endash/></code>	–	#[2013] Strich, etwa ein halbes Geviert lang
<code><emdash/></code>	—	#[2014] Strich, etwa ein ganzes Geviert lang
<code><lquote/></code>	‘	#[2018] Left single quotation mark
<code><rquote/></code>	’	#[2019] Right single quotation mark
<code><bquote/></code>	,	#[201A] Left low single quotation mark (baseline)

Für Sonderzeichen können innerhalb der Definitionen auch eigene Tags definiert werden:

```
<macro name=einhalb>#[00BD]</macro>
```

In diesem Beispiel wird ein Tag mit dem Namen 'einhalb' definiert. Dadurch kann mit dem Tag `<einhalb/>` das Zeichen ½ dargestellt werden.

Die Zeichen `<` und `>` dürfen nur zur Kennzeichnung der Tags, in Akzent-Codierungen (z. B. %<) oder in den Codes für doppelte Anführungszeichen (#.< und #.>) verwendet werden. Sollen die Zeichen `<` und `>` gedruckt werden, können sie (entgegen der Angaben im TUSTEP-Zeichenvorrat) mit `^<` bzw. `^>` codiert werden. Die Zeichen `<` und `>` (die laut TUSTEP-Zeichenvorrat mit `^<` bzw. `^>` codiert werden können) müssen anders codiert werden, z. B. mit #[2329] bzw. #[232A].

Wenn in MS-Word das Anführungszeichen " über Tastatur eingegeben wird, entscheidet MS-Word auf Grund der Umgebung, welches Zeichen dafür eingefügt wird: "Beispiel" wird automatisch zu „Beispiel“. Wenn MS-Word exportierte Daten lädt, erfolgt diese Unterscheidung nicht. Das Zeichen " muss ggf. zuvor durch die Zeichen #. , (für „), #. ' (für “) oder #. " (für ") ersetzt werden. Andernfalls wird das Anführungszeichen so dargestellt: "

Das gleiche gilt, wenn der Apostroph ' über Tastatur eingegeben wird: 'Beispiel' wird automatisch zu ‚Beispiel‘. Wenn MS-Word exportierte Daten lädt, erfolgt diese Unterscheidung nicht. Das Zeichen ' muss ggf. zuvor durch die Zeichen `<bquote/>` (für ,), `<lquote/>` (für ‘) oder `<rquote/>` (für ’) ersetzt werden. Andernfalls wird der Apostroph so dargestellt: '

Beispiel für Anführungszeichen

In diesem Beispiel werden Anfangs- und Ende-Tags definiert, die an Stelle von Anführungszeichen verwendet werden können: Erst ein Text in „deutschen Anführungszeichen“ und dann einer in »französischen Anführungszeichen«.

Außerdem werden Anfangs- und Ende-Tags definiert, die an Stelle von zu druckenden spitzen Klammern verwendet werden können: Ein Text in <spitzen Klammern>. Diese Klammern werden aus einem bestimmten Font genommen, unabhängig davon, welcher Font gerade eingestellt ist.

```
<macro name=anf-d><macro/>#.,<macro/>#.'</macro>
<macro name=anf-f><macro/>#.><macro/>#.<</macro>
```

```
<macro name=spikl>
<macro/><font name="Cambria Math">#[27E8]</font>
<macro/><font name="Cambria Math">#[27E9]</font></macro>
```

```
<p>In diesem Beispiel werden Anfangs- und Ende-Tags definiert, die
an Stelle von Anführungszeichen verwendet werden können: Erst ein
Text in <anf-d>deutschen Anführungszeichen</anf-d> und dann einer
in <anf-f>französischen Anführungszeichen</anf-f>.</p>
```

```
<p>Außerdem werden Anfangs- und Ende-Tags definiert, die an Stelle
von zu druckenden spitzen Klammern verwendet werden können: Ein
Text in <spikl>spitzen Klammern</spikl>. Diese Klammern werden aus
einem bestimmten Font genommen, unabhängig davon, welcher Font
gerade eingestellt ist.</p>
```

Das Tag <macro> ist auf Seite 103 beschrieben.

Silbentrennung

Stellen, an denen ein Wort getrennt werden darf bzw. nicht getrennt werden darf, können mit `\` bzw. mit `\|` gekennzeichnet werden. Letzteres scheint von MS-Word allerdings nicht so interpretiert zu werden.

Mit dem folgenden Tag kann eine Stelle gekennzeichnet werden, an der ein Wort umbrochen werden darf, ohne ein Silbentrennzeichen einzufügen.

`<nospace/>` Optionaler Zeilenwechsel (#[200C])

Für dieses Tag sind keine Attribute vorgesehen.

Wörter, die das Tag `<nospace/>` enthalten, werden nur an dieser Stelle umbrochen oder an einer mit `\` gekennzeichnete Stellen getrennt.

In Absatz-Tags kann mit dem Attribut `hyphenation` (siehe Seite 27) angegeben werden, ob im jeweiligen Absatz Silbentrennung ausgeführt werden soll oder nicht. Im Tag `<dstyle/>` (siehe Seite 19) kann mit dem gleichen Attribut die Voreinstellung dafür angegeben werden.

Soll die Silbentrennung innerhalb eines Absatzes nur für ein oder mehrere Wörter unterdrückt werden, so müssen diese mit einem Zeichenfolgen-Tag, das das Attribut `hyphenation=off` enthält, gekennzeichnet werden. Siehe dazu folgendes Beispiel:

Beispiel zur Silbentrennung

Für diese Beschreibung wurde mit dem am Anfang dieses Beispiels stehenden Tag, das im Ergebnis natürlich nicht zu sehen ist, ein Zeichenfolgen-Tag definiert. Die mit diesem Tag gekennzeichneten Wörter (z. B. `hyphenation`) werden in Courier-Schrift gedruckt und nicht getrennt.

Außerdem wird in diesem Beispiel das Wort Tübingen/Neckar am Zeilenende (Tübingen/Neckar), ohne Silbentrennzeichen umbrochen.

```
<cstyle name=keyword font=Courier hyphenation=off/>
```

```
<p>Für diese Beschreibung wurde mit dem am Anfang dieses Beispiels stehenden Tag, das im Ergebnis natürlich nicht zu sehen ist, ein Zeichenfolgen-Tag definiert. Die mit diesem Tag gekennzeichneten Wörter (z. B. <keyword>hyphenation</keyword>) werden in Courier-Schrift gedruckt und nicht getrennt.</p>
```

```
<p>Außerdem wird in diesem Beispiel das Wort Tübingen/<nospace/>Neckar am Zeilenende (Tübingen/<nospace/>Neckar), ohne Silbentrennzeichen umbrochen. </p>
```

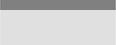
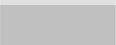
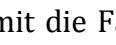
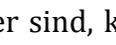
Die Tags `<p>` und `<cstyle/>` sind auf Seite 25 bzw. 101 beschrieben.

Farben

Wenn in dieser Beschreibung für einen Attributwert eine Farbe (c) erwartet wird, muss dafür ein RGB-Wert in hexadezimaler Form `rrggbb` angegeben werden. Um einen solchen Farbwert zu bestimmen, kann z. B. mit Hilfe von MS-Word so vorgegangen werden:

- Ein beliebiges Wort in ein leeres Dokument eintippen.
- Dieses Wort mit der gewünschten Farbe unterlegen.
- Dokument als RTF-Datei (nicht als XML-Datei) abspeichern.
- RTF-Datei nach TUSTEP importieren (mit `IGNORIEREN=-`).
- In der TUSTEP-Datei den Farbwert ablesen.

Für einige Farben sind Namen definiert, die statt eines hexadezimalen Wertes angegeben werden können:

	000000	black		008000	dark-green
	0000FF	blue		77FF33	light-green
	AA3333	brown		FF00FF	magenta
	00FFFF	cyan		800080	dark-magenta
	008080	dark-cyan		800000	maroon
	FFCC00	gold		808000	olive
	C0C0C0	gray		FFAA00	orange
	808080	dark-gray		FFC0CC	pink
	E0E0E0	light-gray		FF0000	red
	C0C0C0	grey		C0C0C0	silver
	808080	dark-grey		9900DD	violet
	E0E0E0	light-grey		FFFFFF	white
	00FF00	green		FFFF00	yellow

Damit die Farbangaben in den Tags sprechender sind, können für Farben innerhalb der Definitionen eigene Namen definiert werden, z. B.:

```
<values hellgrau=E0E0E0 dunkelgrau=C0C0C0 rot=red/>
```

In diesem Beispiel werden die Namen 'hellgrau', 'dunkelgrau' und 'rot' für die entsprechenden Farben definiert. Diese Namen können als Attributwerte verwendet werden z. B.:

```
<p color=hellgrau> . . . </p>
```

Voreinstellungen

Mit dem folgenden Tag können durch Angabe entsprechender Attribute Einstellungen vorgenommen werden, die das ganze Dokument betreffen. Die Einstellungen sind in drei Gruppen unterteilt:

- Einstellungen, die für das gesamte Dokument fix sind.
- Einstellungen, die für jede Absatzgruppe neu festgelegt werden können.
- Einstellungen, die in jedem Absatz geändert werden können.

<dstyle/> (document style / default style)

Das Tag ist optional. Wenn es angegeben wird, dann muss es am Anfang der Definitionen stehen. Davor sind nur Definitionen von Attributwerten mit dem Tag `<values/>` erlaubt. Die Definitionen müssen entweder in der zur Spezifikation DEFINITIONEN angegebenen Datei oder am Anfang der QUELL-Datei vor dem Text stehen (siehe Kapitel "Datenstruktur" Seite 14).

Attribute für Einstellungen, die für das gesamte Dokument fix sind:

<code>template</code>	= <code>pfad</code>	Pfad (mit Dateiname) einer Vorlagendatei.
<code>mirror</code>	= <code>off</code>	Linken und rechten Rand nicht vertauschen. [*]
	= <code>on</code>	Linken und rechten Rand auf Seiten mit gerader Seitennummer vertauschen.
<code>distance</code>	= <code>both</code>	Zwischen zwei Absätzen sowohl den Abstand nach dem vorangehenden Absatz als auch den Abstand vor dem nachfolgenden Absatz berücksichtigen. [*]
	= <code>maximum</code>	Zwischen zwei Absätzen den Abstand vor dem nachfolgenden Absatz um den Abstand nach dem vorangehenden Absatz vermindern, falls im Anfangs-Tag des nachfolgenden nicht explizit das Attribut <code>before</code> angegeben ist.
<code>line-break</code>	= <code>expand</code>	Zeile vor einem Zwangsumbruch mit dem Tag <code>
</code> (siehe Seite 35) ggf. ausgleichen.
<code>tabs</code>	= <code>p</code>	Abstand der Standard-Tabulatoren <code>p</code> Punkt. Die Standard-Tabulatoren werden verwendet, wenn in einem Absatz keine anderen Tabulatoren definiert sind. [18]
<code>footnote</code>	= ...	Angabe, wie Fußnoten nummeriert werden sollen; siehe nachfolgende Aufstellung. [2]
<code>endnote</code>	= ...	Angabe, wie Endnoten nummeriert werden sollen; siehe nachfolgende Aufstellung. [ii]

Mögliche Angaben zu den Attributen `footnote` und `endnote`:

- 1 Absatzgruppenweise mit arabischen Zahlen nummerieren.
- 2 Fortlaufend mit arabischen Zahlen nummerieren.
- i Absatzgruppenweise mit römischen Zahlen in Kleinbuchstaben nummerieren.
- ii Fortlaufend mit römischen Zahlen in Kleinbuchstaben nummerieren.
- I Absatzgruppenweise mit römischen Zahlen in Großbuchstaben nummerieren.

- II Fortlaufend mit römischen Zahlen in Großbuchstaben nummerieren.
- a Absatzgruppenweise mit entsprechenden Kleinbuchstaben nummerieren.
- b Fortlaufend mit entsprechenden Kleinbuchstaben nummerieren.
- A Absatzgruppenweise mit entsprechenden Großbuchstaben nummerieren.
- B Fortlaufend mit entsprechenden Großbuchstaben nummerieren.

Attribute für Einstellungen, die für jede Absatzgruppe neu festgelegt werden können:

<code>left</code>	= <code>p</code>	Linker Rand <code>p</code> Punkt. [65]
<code>right</code>	= <code>p</code>	Rechter Rand <code>p</code> Punkt. [65]
<code>top</code>	= <code>p</code>	Oberer Rand <code>p</code> Punkt. [60]
<code>bottom</code>	= <code>p</code>	Unterer Rand <code>p</code> Punkt. [60]
<code>header</code>	= <code>p</code>	Oberhalb des Kopftextes <code>p</code> Punkt frei. [36]
<code>footer</code>	= <code>p</code>	Unterhalb des Fußtextes <code>p</code> Punkt frei. [36]
<code>pagination</code>	= <code>auto</code>	Seitennummer unten in der Mitte auf jeder Seite. [*]
	= <code>none</code>	Keine Seitennummer automatisch anzeigen.
<code>border</code>	= <code>none</code>	Keinen Rahmen um den Text. [*]
	= <code>text</code>	Rahmen nur um den Text.
	= <code>page</code>	Rahmen um Kopftext, Text und Fußtext.
<code>border-color</code>	= <code>c</code>	Rahmen in Farbe <code>c</code> . [black]
<code>border-width</code>	= <code>p</code>	Rahmen <code>p</code> Punkt dick.
<code>border-space</code>	= <code>p</code>	Zwischen Rahmen und Text (Satzspiegelrand) <code>p</code> Punkt Zwischenraum. [5]

Mit den voreingestellten Werten zu den vier Attributen `left`, `top`, `right` und `bottom` verbleiben für die Satzspiegelbreite 460 Punkt und für die Satzspiegelhöhe 720 Punkt.

Attribute für Einstellungen, die in jedem Absatz geändert werden können:

<code>align</code>	= <code>left</code>	Linksbündig, Flattersatz.
	= <code>center</code>	Auf Mitte zentriert.
	= <code>right</code>	Rechtsbündig, Flattersatz.
	= <code>both</code>	Blocksatz. [*]
	= <code>justify</code>	Wie <code>both</code> .
<code>font</code>	= <code>...</code>	Font name verwenden. [Times]
<code>size</code>	= <code>p</code>	Schriftgröße <code>p</code> Punkt. [12]
<code>kerning</code>	= <code>p</code>	Kerning ab <code>p</code> Punkt.
<code>hyphenation</code>	= <code>on</code>	Mit Silbentrennung. [*]
	= <code>off</code>	Ohne Silbentrennung.

Die Voreinstellungen für Attributwerte sind jeweils in eckigen Klammern angegeben, z. B. [12] beim Attribut `size` für die voreingestellte Schriftgröße 12 Punkt. Sind nur Schlüsselwörter als Attributwert vorgesehen, so ist der voreingestellte Wert mit [*] markiert.

Die Markierung **[**]** wird im Folgenden verwendet, wenn als Voreinstellung der mit dem Tag `<dstyle/>` vorgegebene Wert bzw. dessen Voreinstellung übernommen wird.

Absatzgruppen

Eine Zusammenfassung von Absätzen wird in MS-Word "Abschnitt" (engl. section) genannt. Um eine Verwechslung der Begriffe "Absatz" und "Abschnitt" zu vermeiden, wird in dieser Beschreibung die Zusammenfassung von Absätzen "Absatzgruppe" genannt.

Absätze können zu Absatzgruppen zusammengefasst werden; sie müssen zu Absatzgruppen zusammengefasst werden, wenn Effekte erreicht werden sollen, die nur mit den für Absatzgruppen vorgesehenen Attributen erreicht werden können, oder wenn der Kopf- oder Fußtext gewechselt werden soll.

<section> Absatzgruppe

Für dieses Tag sind folgende Attribute möglich:

page	= same	Keine neue Seite beginnen. [*]
	= next	Neue Seite beginnen.
	= even	Neue Seite mit gerader Seitennummer beginnen.
	= odd	Neue Seite mit ungerader Seitennummer beginnen.
page-number	= decimal	Seitennummer als Dezimalzahl einfügen. [*]
	= roman	Seitennummer als römische Zahl in Kleinbuchstaben.
	= ROMAN	Seitennummer als römische Zahl in Großbuchstaben.
	= n	Seitennummer auf n setzen Statt einer arabischen Zahl kann auch eine römische Zahl in Klein- oder Großbuchstaben angegeben werden; die Seitennummer wird dann in entsprechender Form eingefügt.
line-number	= none	Keine Zeilennummer. [*]
	= n	Zeilen am linken Rand mit der Zeilennummer versehen, wenn die Zeilennummer durch n ohne Rest teilbar ist.
	= n/section	Zeilennummer am Beginn der Absatzgruppe auf 1 zurücksetzen; sonst wie Attributwert n.
	= n/page	Zeilennummer bei jedem Seitenwechsel auf 1 zurücksetzen; sonst wie Attributwert n.
cols	= n	Seiten in n gleich breite Spalten aufteilen. [1]
	= p1 'p2' ...	Seiten in p1, p2, ... Punkt breite Spalten aufteilen.
col-space	= p	Zwischen den Spalten p Punkt Zwischenraum. [18]
col-bar	= yes	Spalten durch einen senkrechten Strich trennen.
	= no	Spalten nicht durch einen Strich trennen. [*]
layout	= portrait	Absatzgruppe im Hochformat drucken. [*]
	= landscape	Absatzgruppe im Querformat drucken.
left	= p	Linker Rand p Punkt. [**]
right	= p	Rechter Rand p Punkt. [**]

<code>top</code>	<code>= p</code>	Oberer Rand <code>p</code> Punkt. [**]
<code>bottom</code>	<code>= p</code>	Unterer Rand <code>p</code> Punkt. [**]
<code>header</code>	<code>= p</code>	Oberhalb des Kopftextes <code>p</code> Punkt frei. [**]
<code>footer</code>	<code>= p</code>	Unterhalb des Fußtextes <code>p</code> Punkt frei. [**]
<code>border</code>	<code>= *</code> <code>= none</code>	Rahmen wie voreingestellt. [**] Keinen Rahmen um den Text.
<code>border-color</code>	<code>= c</code>	Rahmen in Farbe <code>c</code> . [black]
<code>border-width</code>	<code>= p</code>	Rahmen <code>p</code> Punkt dick.
<code>border-space</code>	<code>= p</code>	Zwischen Rahmen und Text (Satzspiegelrand) <code>p</code> Punkt Zwischenraum. [5]
<code>align</code>	<code>= top</code> <code>= center</code> <code>= distribute</code> <code>= bottom</code>	Absätze oben auf der Seite. [*] Absätze in der Mitte der Seite. Absätze verteilt auf der Seite. Absätze unten auf der Seite.
<code>bm</code>	<code>= ...</code>	Name für ein Lesezeichen (siehe Seite 59), das sich auf die ganze Absatzgruppe bezieht.

Die Darstellung der Zeilennummern kann vorgegeben werden, indem ein Tag mit dem Namen 'ln-c' definiert wird, z. B.

```
<cstyle name=ln-c size=10/>
```

Einschränkung: Angaben zum Attribut `align` sind nur auf Seiten wirksam, die nur Absätze derselben Absatzgruppe enthalten.

Beispiel für mehrspaltigen Satz.

Mehrspaltiger Satz ist nicht schwierig. Es muss nur eine neue Absatzgruppe begonnen und die Anzahl der Spalten angegeben werden.

Dieser Text wird drei-
spaltig gedruckt.
Hier sind Absätze und
explizite Zeilenwechsel

genauso möglich wie
bei einspaltigem Druck.
Allerdings ist Flattersatz
empfehlenswert, damit

durch den Randausgleich
zwischen den Wörtern
nicht zu große Lücken
entstehen.

Jetzt geht es wieder mit einspaltigem Satz weiter.

```
<section>
<p>Mehrspaltiger Satz ist nicht schwierig. Es muss nur
eine neue Absatzgruppe begonnen und die Anzahl der Spalten
angegeben werden.</p>
<p/>
</section>
```

```
<section cols=3>
<p align=left>Dieser Text wird dreispaltig gedruckt.</p>
<p align=left>Hier sind Absätze und explizite Zeilenwechsel
<br/>genauso möglich wie<br/>bei einspaltigem Druck.</p>
<p align=left>Allerdings ist Flattersatz empfehlenswert,
damit durch den Randausgleich zwischen den Wörtern nicht zu
große Lücken entstehen.</p>
</section>
```

```
<section>
<p/>
<p>Jetzt geht es wieder mit einspaltigem Satz weiter.</p>
</section>
```

*Die Tags <p> und
 sind auf Seite 25 bzw. 35 beschrieben.*

Absätze

Absatz oder Abschnitt. In dieser Beschreibung wird (wie in MS-Word) der Begriff "Absatz" verwendet, weil der Begriff "Abschnitt" in MS-Word für eine Zusammenfassung von Absätzen verwendet wird.

Neben den in diesem Kapitel beschriebenen Absätzen gibt es auch noch "Listeneinträge" (siehe Kapitel "Listen" Seite 32). Das sind spezielle Absätze, die mit einem Aufzählzeichen (z. B. einem Spiegelstrich) oder einer Nummerierung beginnen und deren Text automatisch links eingezogen wird.

<p> Absatz (paragraph)

Für dieses Tag sind folgende Attribute möglich:

page	= same	Keine neue Seite beginnen. [*]
	= next	Neue Seite beginnen. <i>Hinweis:</i> Eine neue linke bzw. rechte Seite kann mit dem Tag <section> (siehe Seite 22) erreicht werden.
no-break	= within	Innerhalb des Absatzes keine neue Seite beginnen.
	= after	Nach dem Absatz, d. h. zwischen diesem und dem nächsten Absatz keine neue Seite beginnen.
	= title	Wie <i>within</i> und <i>after</i> zusammen.
	= none	Keine Bedingung. [*]
m-pos	= p	Anfang des Absatzes vom oberen Satzspiegelrand aus um mindestens p Punkt nach unten setzen. [0]
first	= p	Die erste Zeile links um p Punkt einrücken. [0]
left	= p	Folgezeilen links um p Punkt einrücken; falls das Attribut <i>first</i> nicht angegeben ist, auch die erste Zeile um p Punkt einrücken. [0]
right	= p	Alle Zeilen rechts um p Punkt einziehen. [0]
direction	= ltr	Left to right: Linken und rechten Einzug wie angeben. [*]
	= rtl	Right to left: Linken und rechten Einzug vertauschen; letzte Zeile des Absatzes ggf. rechtsbündig statt linksbündig.
mirror	= off	Linken und rechten Einzug nicht vertauschen. [*]
	= on	Linken und rechten Einzug auf Seiten mit ungerader Seitennummer vertauschen.
before	= p	Vor dem Absatz zusätzlich p Punkt Abstand. [0]
after	= p	Nach dem Absatz zusätzlich p Punkt Abstand. [0]
distance	= apply	Attribute <i>after</i> und <i>before</i> berücksichtigen. [*]
	= suppress	Falls der vorangehende Absatz mit einem gleichnamigen Absatz-Tag gekennzeichnet ist, wird das Attribut <i>before</i>

		in diesem Absatz ignoriert; falls der nachfolgende Absatz mit einem gleichnamigen Absatz-Tag gekennzeichnet ist, wird das Attribut <code>after</code> in diesem Absatz ignoriert.
<code>align</code>	<code>= left</code>	Linksbündig, Flattersatz.
	<code>= center</code>	Auf Mitte zentrieren.
	<code>= right</code>	Rechtsbündig, Flattersatz.
	<code>= both</code>	Blocksatz. [**]
	<code>= justify</code>	Wie <code>both</code> .
	<code>= distribute</code>	Blocksatz, Randausgleich auch in der Zeile vor einem Zwangsumbruch mit dem Tag <code>
</code> (siehe Seite 35) und in der letzten Zeile des Absatzes.
<code>font</code>	<code>= ...</code>	Font name verwenden. [**]
<code>size</code>	<code>= p</code>	Schriftgröße <code>p</code> Punkt. [**]
<code>kerning</code>	<code>= p</code>	Kerning ab <code>p</code> Punkt. [**]
<code>line-feed</code>	<code>= p</code>	Zeilenabstand mindestens <code>p</code> Punkt.
	<code>= p!</code>	Zeilenabstand genau <code>p</code> Punkt.
<code>scale</code>	<code>= n%</code>	Zeichenbreite auf <code>n</code> Prozent skalieren. [100%]
<code>char-space</code>	<code>= ±p</code>	Zeichenabstand um <code>p</code> Punkt vergrößern/verkleinern. [0]
<code>style</code>	<code>= *</code>	Keine besondere Schriftform. [*]
	<code>= name</code>	Schriftform <code>name</code> , z. B. <code>i</code> für kursiv, <code>b</code> für fett. Für <code>name</code> sind alle Namen der Tags möglich, die in den Kapiteln "Schriftformen" und "Unterstreichung" (siehe Seite 39 bzw. 40) aufgeführt sind. Es können auch mehrere Namen durch Apostroph getrennt angegeben werden.
<code>color</code>	<code>= c</code>	Absatz als Block mit der Farbe <code>c</code> unterlegen (Schattierung).
<code>bgc</code>	<code>= c</code>	Schrift mit der Farbe <code>c</code> unterlegen (Textmarker).
<code>fgc</code>	<code>= c</code>	Schrift in der Farbe <code>c</code> . [black]
<code>ulc</code>	<code>= c</code>	Unterstreichung in der Farbe <code>c</code> . [black]
<code>line-number</code>	<code>= none</code>	Keine Zeilennummer. [*]
<code>border</code>	<code>= none</code>	Keinen Rahmen um den Absatz. [*]
	<code>= box</code>	Rahmen um den Absatz.
	<code>= top</code>	Nur eine Linie vor dem Absatz; z. B. um eine Linie vor dem Fußtext zu ziehen.
	<code>= bottom</code>	Nur eine Linie nach dem Absatz; z. B. um eine Linie nach dem Kopf text zu ziehen.
<code>border-color</code>	<code>= c</code>	Rahmen in Farbe <code>c</code> . [black]
<code>border-width</code>	<code>= p</code>	Rahmen <code>p</code> Punkt dick.

<code>border-space</code>	= <code>p</code>	Zwischen Rahmen und Text <code>p</code> Punkt Zwischenraum. [7]
<code>bar</code>	= <code>c</code>	Absatz am Rand mit einem einfachen senkrechten Strich in der Farbe <code>c</code> versehen.
<code>dblbar</code>	= <code>c</code>	Absatz am Rand mit einem doppelten senkrechten Strich in der Farbe <code>c</code> versehen.
<code>hyphenation</code>	= <code>on</code> = <code>off</code>	Mit Silbentrennung. [*] Ohne Silbentrennung.
<code>codes</code>	= <code>none</code> = <code>name</code>	Keine Zeichen umcodieren. Die in der Code-Tabelle <code>name</code> enthaltenen Zeichen entsprechend umcodieren.
<code>navigation</code>	= <code>none</code> = <code>n</code>	Absatz nicht in die Navigationsleiste einfügen. [*] Text des Absatzes in die Navigationsleiste (siehe Kapitel "Navigation" Seite 59) auf dem Level <code>n</code> einfügen.
<code>next-pstyle</code>	= <code>name</code>	Wenn in MS-Word nach diesem Absatz mit der RETURN-Taste ein neuer Absatz eingefügt wird, soll dieser mit der Formatvorlage <code>name</code> formatiert werden. Diese Formatvorlage muss mit dem Tag <code><pstyle/></code> (siehe Seite 95) definiert werden.
<code>bm</code>	= ...	Name für ein Lesezeichen (siehe Seite 59), das sich auf den ganzen Absatz bezieht.
<code>i-char</code>	= <code>z</code>	Initiale: Buchstabe oder Zeichenfolge <code>z</code> .
<code>i-num</code>	= ...	Initiale: Name der laufende Nummer.
<code>i-font</code>	= ...	Initiale: Font <code>name</code> verwenden.
<code>i-size</code>	= <code>p</code>	Initiale: Schriftgröße <code>p</code> Punkt.
<code>i-space</code>	= <code>p</code>	Initiale: Abstand zum Text <code>p</code> Punkt.
<code>i-bgc</code>	= <code>c</code>	Initiale mit der Farbe <code>c</code> unterlegen.
<code>i-fgc</code>	= <code>c</code>	Initiale in Farbe <code>c</code> . [black]
<code>i-type</code>	= <code>margin</code> = <code>page</code>	Initiale am linken Satzspiegelrand bzw. um den mit dem Attribut <code>left</code> angegebenen Wert eingezogen. [*] Initiale links vom linken Satzspiegelrand.
<code>tabs</code>	= <code>none</code> = <code>t1't2'...</code>	Nur Standard-Tabulatoren. [*] Tabulatordefinitionen, siehe nachfolgenden Absatz.

Eine Tabulatordefinition `t` für das Attribut `tabs` besteht aus einem obligaten und einem optionalen Teil. Falls der optionale Teil angegeben wird, muss er vom obligaten durch einen Doppelpunkt getrennt werden. Der obligate Teil entspricht dem Wert für das Attribut `pos`, der optionale dem Wert für das Attribut `fill` beim Tabulator-Tag `<tab/>` (siehe Seite 53). Zum Beispiel definiert `|72` einen "normalen" (linksbündigen) Tabulator, ein Zoll vom linken Satzspiegelrand entfernt und `360|:.` einen rechtsbündigen Tabulator, fünf Zoll vom linken Satzspiegelrand entfernt, bei dem der entstehende Zwischenraum auspunktet wird.

Der Abstand zwischen zwei Absätzen ergibt sich aus der Summe des Abstandes nach dem jeweils ersten (Attribut `after`) und des Abstandes vor dem jeweils zweiten (Attribut `before`). Mit dem Attribut `distance=maximum` im Tag `<dstyle>` (siehe Seite 19) kann erreicht werden, dass der Abstand insgesamt nur dem größeren der beiden Abstände entspricht.

Bei selbst definierten Tags gilt für das Attribut `style` folgende Regelung: Wird als Attributwert nur ein Stern angegeben oder als erster Teilwert statt eines Namens ein Stern angegeben, so werden die für das Tag voreingestellten Schriftformen übernommen, andernfalls werden sie gelöscht und es werden nur die jeweils angegebenen eingestellt. Beim vordefinierten Tag `<p>` ist die Angabe des Sterns bedeutungslos, da keine Schriftformen voreingestellt sind.

Hinweis: Soll die letzte Zeile eines Absatzes nicht ausgeglichen werden, obwohl `align=distribute` angegeben ist, kann ein Tabulator-Tag `<tab/>` (siehe Seite 40) unmittelbar vor dem Ende-Tag des Absatzes eingefügt werden.

Fehler: Wenn das Attribut `first` zusammen mit dem Attribut `mirror=on` angegeben wird, stimmt in MS-Word auf Seiten mit ungerader Nummer die Einrückung am rechten Rand nicht (Stand Oktober 2010).

Das leere Tag `<p/>` ist gleichbedeutend mit `<p></p>` (ohne Daten zwischen den Tags). Bei selbst definierten Absatz-Tags wird ggf. die in der Definition mit dem Attribut `data` angegebene Zeichenfolge dazwischen eingefügt.

Hinweis: Es gibt keine Möglichkeit, einen Absatz nur dann noch auf einer Seite zu beginnen, wenn noch mindestens `soundso` viel Platz auf der Seite frei ist. Um zu verhindern, dass ein Absatz an einer ungewollten Stelle umbrochen wird, stehen nur die zum Attribut `break` aufgeführten Angaben zur Verfügung.

Beispiel für Absätze

In diesem Absatz wird die erste Zeile eingerückt. Die Folgezeilen werden nicht eingerückt. Die Voreinstellung für die Schriftgröße beträgt 12 Punkt. In den Definitionen kann eine andere Schriftgröße voreingestellt werden.

In diesem Absatz werden alle Zeilen eingerückt und die Schriftgröße wird auf 10 Punkt festgelegt. Wenn solche Absätze mehrfach vorkommen, ist es sinnvoll, in den Definitionen dafür ein eigenes Tag zu definieren, das alle erforderlichen Attribute voreingestellt hat. Dann brauchen die Attribute nicht jedes Mal angegeben zu werden.

```
<p first=18>In diesem Absatz wird die erste Zeile
eingerückt. Die Folgezeilen werden nicht eingerückt. Die
Voreinstellung für die Schriftgröße beträgt 12 Punkt. In den
Definitionen kann eine andere Schriftgröße voreingestellt
werden.</p>
```

```
<p left=18 size=10 before=3 after=3>In diesem Absatz
werden alle Zeilen eingerückt und die Schriftgröße wird auf 10
Punkt festgelegt. Wenn solche Absätze mehrfach vorkommen,
ist es sinnvoll, in den Definitionen dafür ein eigenes Tag zu
definieren, das alle erforderlichen Attribute voreingestellt
hat. Dann brauchen die Attribute nicht jedes Mal angegeben zu
werden.
```

Die Definition eines Tag, wie es im vorangehenden Beispiel empfohlen wird, könnte wie folgt aussehen:

```
<pstyle name=einzug left=18 size=10 before=3 after=3/>
```

Damit wird ein Tag mit dem Namen 'einzug' definiert, das ohne Attribute statt des Tags

```
<p left=18 size=10 before=3 after=3/>
```

verwendet werden kann. In dem selbst definierten Tag <einzug> könnte im Einzelfall auch jedes Attribut angegeben werden, das für das vordefinierte Tag <p> vorgesehen ist; es hätte ggf. Vorrang vor dem in der Definition angegebenen Attribut.

Das selbst definierte Tag <einzug> hat außerdem den Vorteil, dass nur an einer einzigen Stelle (nämlich in der Definition des Tags) geändert werden muss, wenn diese Absätze etwas mehr oder weniger eingezogen werden sollen, oder wenn sie in einer kleineren oder größeren Schrift gedruckt werden sollen. Dies gilt auch für Änderungen in MS-Word, da selbst definierte Absatz-Tags als Formatvorlagen übernommen werden, und dann nur die Formatvorlage geändert werden muss.

Initiale

Wenn in einem Anfangs-Tag für einen Absatz eines der Attribute angegeben ist, deren Name mit `i` beginnt, wird das Zeichen, das unmittelbar auf das Anfangs-Tag folgt, als Initiale vor dem Absatz gedruckt. Dieses Zeichen kann auch mit dem Attribut `i-char` angegeben werden; es muss mit diesem Attribut angegeben werden, wenn das Zeichen, das als Initiale gedruckt werden soll, nicht unmittelbar auf das Anfangs-Tag folgt, oder wenn mehrere Zeichen entsprechend groß am Beginn des Absatzes gedruckt werden soll. Alternativ kann mit dem Attribut `i-num` der Name einer laufenden Nummer angegeben werden; in diesem Fall wird als "Initiale" die entsprechende Nummer gedruckt. Diese Nummer wird in MS-Word jedoch erst angezeigt, nachdem sie aktualisiert wurde, z. B. mit der Tastenkombination Strg+A F9. Die Attribute `i-char` und `i-num` dürfen nicht zusammen angegeben werden.

Beispiel für Initialen

Die Zeilen neben der Initiale werden automatisch der Größe der Initiale entsprechend eingezogen. Ein mit dem Attribut `left` angegebener Einzug gilt auch für die Initiale; in diesem Fall werden die Zeilen neben der Initiale entsprechend weiter eingezogen.

- 1 Als "Initiale" kann nicht nur ein Buchstabe, sondern auch eine Zeichenfolge oder eine laufende Nummer dargestellt werden. In diesem Beispiel wird eine laufende Nummer als "Initiale" auf den linken Seitenrand gedruckt. Passt die Initiale nicht auf den Seitenrand, wird der Text automatisch entsprechend eingerückt.

```
<p i-size=28>Die Zeilen neben der Initiale werden automatisch
der Größe der Initiale entsprechend eingezogen. Ein mit dem
Attribut 'left' angegebener Einzug gilt auch für die Initiale;
in diesem Fall werden die Zeilen neben der Initiale entsprechend
weiter eingezogen.</p>
```

```
<p/>
```

```
<p i-size=28 i-num=init i-type=page i-space=10>Als "Initiale"
kann nicht nur ein Buchstabe, sondern auch eine Zeichenfolge
oder eine laufende Nummer dargestellt werden. In diesem Beispiel
wird eine laufende Nummer als "Initiale" auf den linken
Seitenrand gedruckt. Passt die Initiale nicht auf den Seitenrand,
wird der Text automatisch entsprechend eingerückt.</p>
```

Beispiel für Rahmen und Schattierungen

Ein Absatz mit einem einfachen Strich als Markierung am äußeren Seitenrand. |

Ein Absatz mit einem doppelten Strich als Markierung am äußeren Seitenrand. ||

Ein Absatz mit silbergrauem Textmarker.

Ein Absatz mit einem Rahmen in voller Breite.

Ein Absatz in silbergrau in voller Breite, ohne Rahmen.

Ein Absatz in silbergrau in voller Breite, mit Rahmen.

Ein Absatz in silbergrau, eingezogen, mit Rahmen.

```

<p bar=black>Ein Absatz mit einem einfachen
Strich als Markierung am äußeren Seitenrand.</p>
<p/>
<p dblbar=black>Ein Absatz mit einem doppelten
Strich als Markierung am äußeren Seitenrand.</p>
<p/>
<p bgc=silver>Ein Absatz mit silbergrauem Textmarker.</p>
<p/>
<p border=box>Ein Absatz mit einem Rahmen
in voller Breite.</p>
<p/>
<p color=silver>Ein Absatz in silbergrau
in voller Breite, ohne Rahmen.</p>
<p/>
<p color=silver border=box>Ein Absatz in silbergrau
in voller Breite, mit Rahmen.</p>
<p/>
<p color=silver border-color=silver left=72 right=72 align=center>
Ein Absatz in silbergrau, eingezogen, mit Rahmen.</p>

```

Listen

Listen werden auch "Aufzählungen", "Aufzählungslisten" oder "Nummerierte Listen" genannt. Sie enthalten Listeneinträge. Ein Listeneintrag ist ein spezieller Absatz, der mit einem Aufzählzeichen (z. B. einem Spiegelstrich) oder einer Nummerierung beginnt, und dessen Text automatisch links eingezogen wird.

**** Listeneintrag (list item)

Als Aufzählzeichen dient ein ausgefüllter Kreis, ab dem zweiten Level ein Gedankenstrich. Der Text wird jeweils um einen Standard-Tabulator weiter eingezogen. Andere Listenformate (z. B. nummerierte Listen) können innerhalb der Definitionen mit dem Tag `<lstyle>` (siehe Seite 97) definiert werden.

Hinweis: Die Voreinstellung für den Abstand der Standard-Tabulatoren beträgt 18 Punkt. Sie kann mit dem Tag `<dstyle/>` (siehe Seite 19) festgelegt werden.

Mit Ausnahme der Attribute `first`, `left` und `direction` sowie den Attributen, deren Name mit dem Buchstaben `i` beginnt, können alle Attribute angegeben werden, die für "normale" Absätze (siehe Seite 25) vorgesehen sind.

Zusätzlich sind die beiden folgenden Attribute möglich:

<code>level</code>	= <code>n</code>	Für mehrstufige Listen gibt <code>n</code> den Listen-Level an. Der gesamte Absatz wird automatisch um <code>n-1</code> Standard-Tabulatoren eingerückt, ggf. zusätzlich zu dem mit dem Attribut <code>indent</code> angegebenen Einzug. [1]
<code>indent</code>	= <code>p</code>	Alle Zeilen links zusätzlich zu dem vom Attribut <code>level</code> abhängigen Einzug um <code>p</code> Punkt einziehen. [0]

Anfang und Ende einer Liste müssen für den Export nicht gekennzeichnet sein. Sind sie z. B. mit den Tags `<liste>` und `</liste>` gekennzeichnet, so können diese beiden Tags innerhalb der Definitionen mit

```
<alias alias=liste tag=dummy/>
```

als zu ignorierende Tags definiert werden (siehe Kapitel "Tags ignorieren" Seite 52).

Beispiel für Listen

- Der erste Listeneintrag ist etwas länger als der zweite, damit man sieht, dass der Text der Folgezeilen genauso wie der Text der ersten Zeile eingezogen wird.
 - Der zweite Listeneintrag ist kurz, hat aber zwei Untereinträge.
 - Erster Untereintrag
 - Noch ein Untereintrag
 - Zum Schluss wieder ein Haupteintrag.
-

```
<li>Der erste Listeneintrag ist etwas länger als der zweite,  
damit man sieht, dass der Text der Folgezeilen genauso wie  
der Text der ersten Zeile eingezogen wird.</li>
```

```
<li>Der zweite Listeneintrag ist kurz, hat aber  
zwei Untereinträge.</li>
```

```
<li level=2>Erster Untereintrag</li>
```

```
<li level=2>Noch ein Untereintrag</li>
```

```
<li>Zum Schluss wieder ein Haupteintrag.</li>
```

Trennlinien

Zwischen Absätzen kann eine waagrechte Linie eingefügt werden. Die Länge und der Abstand zum vorangehenden und nachfolgenden Absatz können durch Attribute festgelegt werden.

`<hr/>` Trennlinie (horizontal rule)

Für dieses Tag sind folgende Attribute möglich:

<code>left</code>	= <code>p</code>	Trennlinie <code>p</code> Punkt vom linken Rand einziehen. [0]
<code>right</code>	= <code>p</code>	Trennlinie <code>p</code> Punkt vom rechten Rand einziehen. [0]
<code>no-break</code>	= <code>after</code>	Zwischen der Trennlinie und dem nächsten Absatz keine neue Seite beginnen.
	= <code>none</code>	Keine Bedingung. [*]
<code>mirror</code>	= <code>off</code>	Linken und rechten Einzug nicht vertauschen. [*]
	= <code>on</code>	Linken und rechten Einzug auf Seiten mit ungerader Seitennummer vertauschen.
<code>before</code>	= <code>p</code>	Vor der Trennlinie zusätzlich <code>p</code> Punkt Abstand.
<code>after</code>	= <code>p</code>	Nach der Trennlinie zusätzlich <code>p</code> Punkt Abstand.
<code>line-color</code>	= <code>c</code>	Trennlinie in der Farbe <code>c</code> . [black]
<code>line-width</code>	= <code>p</code>	Trennlinie <code>p</code> Punkt dick.

Hinweis: Senkrechte Trennlinien zwischen den Spalten bei mehrspaltigem Text können mit dem Tag `<ssstyle>` (siehe Seite 94) verlangt werden.

Zwangsumbruch

Innerhalb eines Absatzes kann mit folgendem Tag eine neue Zeile, Spalte oder Seite begonnen werden; die nachfolgende Zeile wird ggf. wie eine Folgezeile eingerückt.

**
** Umbruch (break)

Für dieses Tag ist folgendes Attribut möglich:

type	= line	Zeilenwechsel. [*]
	= column	Spaltenwechsel.
	= page	Seitenwechsel.

Für Absätze mit Blocksatz auf Grund des Attributs `align=both` (siehe Seite 26) gilt: Die Zeile vor einem Zwangsumbruch wird ausgeglichen, wenn im Tag `<dstyle/>` (siehe Seite 19) das Attribut `line-break=expand` angegeben ist, andernfalls wird sie nicht ausgeglichen.

Hinweis: Soll die Zeile vor einem Zwangsumbruch im Einzelfall nicht ausgeglichen werden, obwohl `line-break=expand` angegeben ist, kann ein Tabulator-Tag `<tab/>` (siehe Seite 53) unmittelbar vor dem Tag `
` eingefügt werden.

Soll bei jedem Zeilenwechsel in den Ausgangsdaten auch im Ergebnis eine neue Zeile begonnen werden, soll kann dies mit folgendem Tag erreicht werden:

**
** Automatischer Zeilenwechsel

Für dieses Tag sind keine Attribute vorgesehen.

Innerhalb eines mit `
` und `</br>` gekennzeichneten Bereichs wird bei jedem Zeilenwechsel in den Ausgangsdaten automatische das Tag `
` ergänzt. Der Zeilenabstand kann ggf. mit dem Attribut `line-feed` im jeweils aktuellen Absatz-Tag vorgegeben werden.

Zeichenfolgen

Die in diesem Kapitel beschriebenen Tags können verwendet werden, um innerhalb eines Absatzes einzelne Zeichenfolgen verschieden darzustellen.

<cs> Zeichenfolge (character string)

Für dieses Tag sind folgende Attribute möglich:

<code>font</code>	= ...	Font name verwenden.
<code>size</code>	= p	Schriftgröße p Punkt.
	= ±p	Schriftgröße um p Punkt vergrößern/verkleinern.
<code>kerning</code>	= p	Kerning ab p Punkt.
<code>scale</code>	= n%	Zeichenbreite auf n Prozent skalieren. [100%]
<code>char-space</code>	= ±p	Zeichenabstand um p Punkt vergrößern/verkleinern. [0]
<code>width</code>	= p	Zeichenfolge auf p Punkt einpassen.
<code>style</code>	= *	Keine besondere Schriftform. [*]
	= name	Schriftform name, z. B. <i>i</i> für kursiv, b für fett. Für name sind alle Namen der Tags möglich, die in den Kapiteln "Schriftformen" und "Unterstreichung" (siehe Seite 39 bzw. 40) aufgeführt sind. Es können auch mehrere Namen durch Apostroph getrennt angegeben werden.
<code>bgc</code>	= c	Schrift mit der Farbe c unterlegen (Textmarker).
<code>fgc</code>	= c	Schrift in der Farbe c.
<code>ulc</code>	= c	Unterstreichung in der Farbe c. [black]
<code>border</code>	= none	Keinen Rahmen um die Zeichenfolge. [*]
	= box	Rahmen um die Zeichenfolge.
<code>border-color</code>	= c	Rahmen in Farbe c. [black]
<code>border-width</code>	= p	Rahmen p Punkt dick.
<code>border-space</code>	= p	Zwischen Rahmen und Text p Punkt Zwischenraum. [7]
<code>up</code>	= p	Text um p Punkt hochstellen.
<code>dn</code>	= p	Text um p Punkt tiefstellen.
<code>hyphenation</code>	= off	Keine Silbentrennung.
<code>codes</code>	= none	Keine Zeichen umcodieren.
	= name	Die in der Code-Tabelle name enthaltenen Zeichen entsprechend umcodieren.

Statt des zuvor beschriebenen Tags `<cs>` werden in importierten Daten die nachfolgend beschriebenen Tags ``, `<color>` und `<pos>` verwendet; diese Tags können aber auch beim Exportieren verwendet werden.

**** Schriftart

Für dieses Tag sind folgende Attribute möglich:

name	= name	Font name verwenden.
size	= p	Schriftgröße p Punkt.
	= ±p	Schriftgröße um p Punkt vergrößern/verkleinern.
kerning	= p	Kerning ab p Punkt.
scale	= n%	Zeichenbreite auf n Prozent skalieren. [100%]
char-space	= ±p	Zeichenabstand um p Punkt vergrößern/verkleinern. [0]
width	= p	Zeichenfolge auf p Punkt einpassen.
style	= *	Schriftform nicht ändern. [*]
	= name	Schriftform name, z. B. i für kursiv, b für fett. Für name sind alle Namen der Tags möglich, die in den Kapiteln "Schriftformen" und "Unterstreichung" (siehe Seite 39 bzw. 40) aufgeführt sind. Es können auch mehrere Namen durch Apostroph getrennt angegeben werden.
up	= p	Text um p Punkt hochstellen.
dn	= p	Text um p Punkt tiefstellen.
codes	= none	Keine Zeichen umcodieren.
	= name	Die in der Code-Tabelle name enthaltenen Zeichen entsprechend umcodieren.

Beispiel für Zeichenbreite und Zeichenabstand

Mit dem Attribut 'scale' können die einzelnen Zeichen schmaler oder breiter gedruckt werden. Die Zeichenhöhe bleibt dabei unverändert.

Mit dem Attribut 'char-space' kann der Abstand zwischen den einzelnen Zeichen verkleinert bzw. vergrößert werden. Die Zeichenbreite bleibt dabei unverändert.

```

<p>Mit dem Attribut 'scale' können die einzelnen Zeichen
<font scale="90%">schmaler</font> oder
<font scale="120%">breiter</font> gedruckt werden.
Die Zeichenhöhe bleibt dabei unverändert.</p>
<p>Mit dem Attribut 'char-space' kann
der Abstand zwischen den einzelnen Zeichen
<font char-space=-0.8>verkleinert</font> bzw.
<font char-space=+1.2>vergrößert</font> werden.
Die Zeichenbreite bleibt dabei unverändert.</p>

```

<color> Farbe

Für dieses Tag sind folgende Attribute möglich:

<code>bgc</code>	= <code>c</code>	Schrift mit der Farbe <code>c</code> unterlegen (Textmarker).
<code>fgc</code>	= <code>c</code>	Schrift in der Farbe <code>c</code> .
<code>ulc</code>	= <code>c</code>	Unterstreichung in der Farbe <code>c</code> . [black]
<code>border</code>	= <code>none</code> = <code>box</code>	Keinen Rahmen um die Zeichenfolge. [*] Rahmen um die Zeichenfolge.
<code>border-color</code>	= <code>c</code>	Rahmen in Farbe <code>c</code> . [black]
<code>border-width</code>	= <code>p</code>	Rahmen <code>p</code> Punkt dick.
<code>border-space</code>	= <code>p</code>	Zwischen Rahmen und Text <code>p</code> Punkt Zwischenraum. [7]

Beispiel für einen Textmarker und einen Textrahmen

So sieht ein **markierter Text** und so ein **eingerahmter Text** aus.

```
<p>So sieht ein <color bgc=yellow>markierter Text</color>
und so eine <color border=box>eingerahmter Text</color>
aus.</p>
```

<pos> Positionieren

Für dieses Tag sind folgende Attribute möglich:

<code>up</code>	= <code>p</code>	Text um <code>p</code> Punkt hochstellen.
<code>dn</code>	= <code>p</code>	Text um <code>p</code> Punkt tiefstellen.
<code>size</code>	= <code>p</code>	Schriftgröße <code>p</code> Punkt.
<code>hyphenation</code>	= <code>off</code>	Keine Silbentrennung.

Beispiel für Hoch- und Tiefstellung

Text, der mit den Attributen 'up' oder 'dn' **hochgestellt** bzw. **tiefgestellt** ist, wird in unveränderter Schriftgröße gedruckt, falls die Schriftgröße nicht explizit mit dem Attribut 'size' angegeben wird. Bei Hoch- und Tiefstellung mit den Tags `<sup>` und `<sub>` wird dagegen automatisch eine kleinere Schrift verwendet.

```
<p>Text, der mit den Attributen 'up' oder 'dn'
<pos up=4>hochgestellt</pos> bzw. <pos dn=2>tiefgestellt</pos>
ist, wird in unveränderter Schriftgröße gedruckt, falls
die Schriftgröße nicht explizit mit dem Attribut 'size'
angegeben wird. Bei Hoch- und Tiefstellung mit den Tags
<sup>^<sup^></sup> und <sub>^<sub^></sub> wird dagegen
automatisch eine kleinere Schrift verwendet.</p>
```

Schriftformen

Die in diesem Kapitel beschriebenen Tags können verwendet werden, um innerhalb eines Absatzes einzelne Zeichenfolgen verschieden darzustellen.

Hinweis: Um den ganzen Absatz in einer bestimmten Schriftform darzustellen, ist kein eigenes Tag erforderlich: der Name des Tags kann zum Attribut `style` im Anfangs-Tag des jeweiligen Absatzes angegeben werden.

Die Wirkung eines Anfangs-Tags wird mit dem entsprechenden Ende-Tag wieder aufgehoben. Wenn aber z. B. im Anfangs-Tag eines Absatzes mit dem Attribut `style=i` Kursivdruck verlangt wurde und innerhalb des Absatzes ein Wort nicht kursiv gedruckt werden soll, wird ein Tag benötigt, das den Kursivdruck aufhebt. Die für solche Zwecke benötigten Tags sind in den folgenden Aufstellungen jeweils am rechten Rand für das am linken Rand stehende Tag angegeben.

Beispiel für Aufhebungen

In diesem Beispiel soll das Wort TUSTEP nicht kursiv gedruckt werden.

```
<p style=i>In diesem Beispiel soll das Wort <ni>TUSTEP</ni>
nicht kursiv gedruckt werden.</p>
```

Für diese Tags sind keine Attribute vorgesehen.

	Beispieltext	fett (bold)	<nb>
<c>	BEISPIELTEXT	Kapitälchen (small caps)	<nc>
<C>	BEISPIELTEXT	Versalien (capital letters)	<nC>
<i>	<i>Beispieltext</i>	kursiv (italic)	<ni>
<s>	Beispieltext	gesperrt (stretched)	<ns>
<h>		verborgen (hidden)	<nh>
<E>	Beispieltext	Relief (embossed)	<nE>
<I>	Beispieltext	Gravur (imprinted)	<nI>
<O>	Beispieltext	Umriss (outlined)	<nO>
<S>	Beispieltext	schattiert (shadowed)	<nS>
<sub>	Beispieltext	tiefgestellt, kleinere Schrift	<bl>
<sup>	Beispieltext	hochgestellt, kleinere Schrift	<bl>

Folgende TUSTEP-Codierungen können alternativ zu Tags verwendet werden:

#/+	<i>Beispieltext</i>	kursiv
#f+	Beispieltext	fett
#k+	BEISPIELTEXT	Kapitälchen
#v+	BEISPIELTEXT	Versalien
#s+	Beispieltext	gesperrt

Hinweis: Soll z. B. das Tag <k> die gleiche Wirkung haben wie das Tag <i>, kann es mit

```
<alias alias=k tag=i/>
```

definiert und alternativ zum Tag <i> für kursiv verwendet werden.

Unterstreichung

Die in diesem Kapitel beschriebenen Tags können verwendet werden, um innerhalb eines Absatzes einzelne Zeichenfolgen zu unterstreichen oder durchzustrichen.

In der folgenden Aufstellung ist am rechten Rand jeweils das Tag angegeben, mit dem das am linken Rand stehende Tag aufgehoben werden kann. In welchem Fall diese am rechten Rand angegebenen Tags erforderlich sind, ist am Anfang des vorangehenden Kapitels "Schriftformen" auf Seite 39 beschreiben.

Bei diesen Tags kann mit dem Attribut `ulc` die Farbe für die Unterstreichung angegeben werden, z. B. `<u1 ulc=red>` damit rot unterstrichen wird.

<code><u1></code>	<u>Beispieltext</u>	einfach unterstrichen (underlined)	<code><nu>,<nu1></code>
<code><u2></code>	<u><u>Beispieltext</u></u>	doppelt unterstrichen (underlined)	<code><nu>,<nu2></code>
<code><ut></code>	<u>Beispieltext</u>	dick unterstrichen (underlined, thick)	<code><nu>,<nut></code>
<code><s1></code>	Beispieltext	einfach durchgestrichen (striked)	<code><ns1></code>
<code><s2></code>	<u>Beispieltext</u>	doppelt durchgestrichen (striked)	<code><ns2></code>
<code><df></code>	<u>Beispieltext</u>	dünn gestrichelt (dashed, fine)	<code><ndf></code>
<code><dt></code>	<u>Beispieltext</u>	dick gestrichelt (dashed, thick)	<code><ndt></code>
<code><Df></code>	<u>Beispieltext</u>	dünn gestrichelt-punktiert (dash-dotted, fine)	<code><nDf></code>
<code><Dt></code>	<u>Beispieltext</u>	dick gestrichelt-punktiert (dash-dotted, thick)	<code><nDt></code>
<code><lf></code>	<u>Beispieltext</u>	lang dünn gestrichelt (long dashed, fine)	<code><nlf></code>
<code><lt></code>	<u>Beispieltext</u>	lang dick gestrichelt (long dashed, thick)	<code><nlt></code>
<code><of></code>	<u>Beispieltext</u>	dünn unterpunktiert (dotted, fine)	<code><nof></code>
<code><ot></code>	<u>Beispieltext</u>	dick unterpunktiert (dotted, thick)	<code><not></code>
<code><Of></code>	<u>Beispieltext</u>	dünn punktiert-gestrichelt (dot-dash-dotted, fine)	<code><nOf></code>
<code><Ot></code>	<u>Beispieltext</u>	dick punktiert-gestrichelt (dot-dash-dotted, thick)	<code><nOt></code>
<code><uw></code>	<u>Beispieltext</u>	wortweise unterstrichen (underlined, word)	<code><nu>,<nuw></code>
<code><w1></code>	<u>Beispieltext</u>	einfach gewellt unterstrichen (waved)	<code><nu>,<nw1></code>
<code><w2></code>	<u>Beispieltext</u>	doppelt gewellt unterstrichen (waved)	<code><nu>,<nw2></code>
<code><wt></code>	<u>Beispieltext</u>	dick gewellt unterstrichen (waved, thick)	<code><nu>,<nwt></code>

Folgende TUSTEP-Codierungen können alternativ zu Tags verwendet werden:

#0+	<u>Beispieltext</u>	durchgestrichen
#1+	<u>Beispieltext</u>	einfach unterstrichen
#2+	<u><u>Beispieltext</u></u>	doppelt unterstrichen
#3+	<u>Beispieltext</u>	dick unterstrichen
#4+	<u>Beispieltext</u>	unterpunktiert
#7+	<u>Beispieltext</u>	helle Unterlegung
#8+	<u>Beispieltext</u>	dunkle Unterlegung

Schriftumschaltung

Schriftumschaltungen sind mit folgenden TUSTEP-Codierungen möglich:

#a+	ابتجخسشصضطظع	Arabisch
#g+	αβγδεζηθικλμνξοπ	Griechisch
#h+	צפגסגמלכלטיחזחיהגדבא	Hebräisch
#p+	iiɪiγyυσϖüωœεεэ	Phonetisch
#r+	абвгдеёжзийкклмн	Russisch

Schriftumschaltungen sind nur sinnvoll, wenn der verwendete Font ein Unicode-Font ist und die jeweiligen Schriftzeichen enthält!

Hinweis: Auch für diese Schriftumschaltungen können eigene Tags definiert werden, z. B.

```
<cstyle name="hebr" start="#h+" end="#h-"/>
```

Damit kann das Tag <hebr> zum Umschalten auf hebräische Schrift verwendet werden.

Hebräische Buchstaben werden in TUSTEP mit lateinischen Kleinbuchstaben, hebräische Buchstaben mit Dagesch mit lateinischen Großbuchstaben codiert. Beim Exportieren werden erstere in den Unicode-Bereich 05D0 bis 05EA, letztere in den Bereich FB30 bis FB4A umcodiert. Da sich in manchen Fonts die Buchstaben aus dem Bereich FB30 bis FB40 nicht mit weiteren Punktierungszeichen kombinieren lassen, können auch die Großbuchstaben in den Bereich 05D0 bis 05EA umcodiert und mit einem Code für den Dagesch versehen werden. Dazu muss eine Codetabelle (siehe Seite 105) erstellt und dabei der Code für den Dagesch (in der Regel mit 05BC=dagesh) angegeben werden. Ob es sinnvoll ist, die Buchstaben mit Dagesch in den Bereich 05D0 bis 05EA umzucodieren, hängt vom verwendeten Font ab, da in diesem Fall der Dagesch nicht immer optimal in den Buchstaben positioniert wird.

Die obige Definition des Tags zur Schriftumschaltung kann ggf. wie folgt ergänzt werden:

```
<codes name=hebcode 05BC=dagesh/>
```

```
<cstyle name="hebr" start="#h+" end="#h-" codes=hebcod/>
```

Fußnoten, Endnoten

Wenn mit dem Tag `<dstyle/>` (siehe Seite 19) nichts anderes festgelegt wird, werden Fußnoten unten auf die jeweilige Seite, Endnoten am Ende des Textes gedruckt. Fußnoten werden mit arabischen, Endnoten mit römischen Zahlen fortlaufend nummeriert.

Soll mit jedem Kapitel die Numerierung der Fußnoten neu beginnen, so muss jedes Kapitel als Absatzgruppe (siehe Seite 22) mit den Tags `<section>` und `</section>` gekennzeichnet werden und im Tag `<dstyle/>` ein entsprechender Wert zum Attribut `footnote` angegeben werden.

`<fn>` Fußnote

`<en>` Endnote

Für beide Tags sind folgende Attribute möglich:

<code>char</code>	= <code>z</code>	Fußnoten-Nummer bzw. Endnoten-Nummer nicht erhöhen und statt einer Nummer das Zeichen <code>z</code> verwenden.
<code>auto-tabs</code>	= <code>before</code>	In der Fußnote bzw. Endnote am Anfang vor der Nummer einen Tabulator einfügen.
	= <code>after</code>	In der Fußnote bzw. Endnote zwischen Nummer und Text einen Tabulator einfügen. [*]
	= <code>n</code>	Im zweiten Absatz und in den folgenden Absätzen an Stelle der Fußnotennummer bzw. Endnotennummer insgesamt <code>n</code> Tabulatoren einfügen. [1]
<code>bm</code>	= ...	Name für ein Lesezeichen (siehe Seite 59), das sich auf die ganze Fußnote bzw. Endnote bezieht.

Der Fußnotentext kann aus einem oder mehreren Absätzen bestehen. Wenn er nur aus einem Absatz besteht, können das Anfangs-Tag und das Ende-Tag für den Absatz weggelassen werden. In diesem Fall werden intern die Tags `<fn-p>` und `</fn-p>` ergänzt. Das Tag `<fn-p>` ist vordefiniert, wobei die Attribute `font`, `size` und `align` aus dem Tag `<dstyle/>` (siehe Seite 19) bzw. dessen Voreinstellungen übernommen werden. Die Schriftgröße wird jedoch um 1 Punkt reduziert, sodass die Fußnoten um 1 Punkt kleiner gedruckt werden als der Text. Außerdem wird der Fußnotentext um 18 Punkt eingezogen. Das Tag `<fn-p>` kann innerhalb der Definitionen auch selbst definiert werden, z. B.

```
<pstyle name=fn-p font=Times size=10 first=0 left=18/>
```

Für den Endnotentext gilt dasselbe, jedoch wird das Tag `<en-p>` verwendet.

Die Darstellung der Fußnotennummern kann vorgegeben werden, indem ein Zeichenfolgen-Tag mit dem Namen 'fn-c' definiert wird, z. B.

```
<cstyle name=fn-c style=sup'ni'nb size=10/>
```

Damit werden alle Fußnotennummern (sowohl im Text als auch vor den Fußnoten) mit hochgestellten Zeichen passend zur Schriftgröße 10 Punkt, nicht kursiv und nicht fett dargestellt. Ohne diese Definition werden die Fußnotennummern entsprechend der jeweils aktuellen Schriftgröße und ggf. kursiv oder fett dargestellt.

Für die Endnotennummern gilt dasselbe, jedoch wird das Tag `<en-c>` verwendet.

Beispiel für Fußnoten

Text mit Fußnoten.¹ Zur Demonstration eine Fußnote, die nicht mitgezählt wird.* Eigentlich ist nur ein Ersatzzeichen vorgesehen, aber MS-Word erlaubt auch bis zu drei Sternchen. Zum Schluss wieder eine normale Fußnote.²

Die sich aus dem Beispiel ergebenden Fußnoten stehen unten auf dieser Seite.

```
<p>Text mit Fußnoten.<fn>Fußnote mit der Nummer eins.</fn>
Zur Demonstration eine Fußnote, die nicht mitgezählt wird.
<fn char="*">Sternchen-Fußnote.</fn>
Eigentlich ist nur ein Ersatzzeichen vorgesehen, aber MS-Word
erlaubt auch bis zu drei Sternchen. Zum Schluss wieder eine
normale Fußnote.<fn>Fußnote mit der Nummer zwei.</fn></p>
```

In Word werden die Fußnotennummern üblicherweise linksbündig am Satzspiegelrand gedruckt. Sollen sie rechtsbündig gedruckt werden, so dass die Neun der Fußnotennummer 9 exakt über der Null der Fußnotennummer 10 steht, muss dafür ein rechtsbündiger Tabulator in dem für Fußnoten verwendeten Absatz-Tag gesetzt werden. Zusätzlich muss im Anfangs-Tag für die Fußnote mit dem Attribut `auto-tabs` angegeben werden, dass auch vor der Fußnotennummer ein Tabulator eingefügt werden soll.

Besteht der Fußnotentext/Endnotentext aus mehreren Absätzen und sollen beim zweiten und den folgenden Absätzen die erste Zeile weiter als die anderen Zeilen eingezogen werden, muss entweder im Anfangs-Tag für den Absatz ein weiterer Tabulator an die entsprechende Position gesetzt und im Anfangs-Tag für die Fußnote mit dem Attribut `auto-tabs` angegeben werden, dass ein zusätzlicher Tabulator an Stelle der Fußnotennummer eingefügt werden soll, oder es muss beim zweiten und den folgenden Absätzen ein Absatz-Tag verwendet werden, bei dem mit dem Attribut `first` angegeben ist, wie weit die erste Zeile eingerückt werden soll, und im Anfangs-Tag für die Fußnote muss mit dem Attribut `auto-tabs` angegeben werden, dass kein Tabulator an Stelle der Fußnotennummer eingefügt werden soll. Im nachfolgenden Beispiel wird die erste Variante verwendet.

¹ Fußnote mit der Nummer eins.

* Sternchen-Fußnote.

² Fußnote mit der Nummer zwei.

Weiteres Beispiel für Fußnoten

Zuerst einige Fußnoten³, damit sich die Fußnotennummer⁴ soweit erhöht, dass bei der neunten/zehnten Fußnote⁵ zu sehen ist, dass die Nummern⁶ wie gewünscht rechtsbündig übereinander stehen. Nun ein Text mit eins⁷, zwei⁸, drei⁹, vier¹⁰ weiteren Fußnoten.

Die sich aus dem Beispiel ergebenden Fußnoten stehen unten auf dieser Seite.

```
<pstyle name=fn-p size=10 first=0 left=18 tabs=8|'|30/>
```

```
<alias alias=fn tag=fn auto-tabs=before'after'2/>
<alias alias=fn-oe tag=fn auto-tabs=before'after'2/>
<alias alias=fn-me tag=fn auto-tabs=before'after'3/>
```

```
<p>Zuerst einige Fußnoten<fn>Fußnote</fn>, damit sich
die Fußnotennummer<fn>Fußnote</fn> soweit erhöht, dass
bei der neunten/zehnten Fußnote<fn>Fußnote</fn> zu sehen ist,
dass die Nummern<fn>Fußnote</fn> wie gewünscht
rechtsbündig übereinander stehen. Nun ein Text mit
eins<fn-oe>Einziger Absatz der Fußnote;
das Absatz-Tag <fn-p> wird automatisch ergänzt.</fn-oe>,
zwei<fn-oe><fn-p>Einziger Absatz der Fußnote;
das Absatz-Tag <fn-p> ist angegeben.</fn-p></fn-oe>,
drei<fn-oe><fn-p>Erster Absatz der Fußnote; das Absatz-Tag
muss angegeben werden.</fn-p><fn-p>Zweiter Absatz der
Fußnote; das Absatz-Tag muss angegeben werden. Beim
zweiten und den folgenden Absätzen wird die erste Zeile
nicht weiter eingerückt.</fn-p></fn-oe>,
vier<fn-me><fn-p>Erster Absatz der Fußnote; das Absatz-Tag
muss angegeben werden. Der erste Absatz einer Fußnote mit
mehreren Absätzen wird genauso dargestellt, wie wenn die
Fußnote nur einen Absatz hätte.</fn-p><fn-p>Zweiter Absatz
der Fußnote; das Absatz-Tag muss angegeben werden. Beim
zweiten und den folgenden Absätzen wird jeweils die erste
Zeile weiter eingerückt.</fn-p></fn-me> weiteren Fußnoten.</p>
```

Das Tag <alias> ist auf Seite 92 beschrieben.

³ Fußnote

⁴ Fußnote

⁵ Fußnote

⁶ Fußnote

⁷ Einziger Absatz der Fußnote; das Absatz-Tag <fn-p> wird automatisch ergänzt.

⁸ Einziger Absatz der Fußnote; das Absatz-Tag <fn-p> ist angegeben.

⁹ Erster Absatz der Fußnote; das Absatz-Tag muss angegeben werden.

Zweiter Absatz der Fußnote; das Absatz-Tag muss angegeben werden. Beim zweiten und den folgenden Absätzen wird die erste Zeile nicht weiter eingerückt.

¹⁰ Erster Absatz der Fußnote; das Absatz-Tag muss angegeben werden. Der erste Absatz einer Fußnote mit mehreren Absätzen wird genauso dargestellt, wie wenn die Fußnote nur einen Absatz hätte.

Zweiter Absatz der Fußnote; das Absatz-Tag muss angegeben werden. Beim zweiten und den folgenden Absätzen wird jeweils die erste Zeile weiter eingerückt.

Anmerkungen

Anmerkungen können wie Fußnoten in den Text eingefügt werden. Der Anmerkungstext wird an den entsprechenden Stellen in einem Rahmen am rechten Rand eingeblendet. Eine Anmerkung kann sich auf eine Stelle im Text oder auf einen Textbereich beziehen. Von jedem Anmerkungstext führt eine gestrichelte Linie zu der entsprechenden Stelle im Text bzw. ans Ende des entsprechenden Textbereichs. Wenn sich eine Anmerkung auf einen Textbereich bezieht, wird dieser Bereich in MS-Word hervorgehoben, wenn mit der Maus auf den Anmerkungstext geklickt wird.

<an> Anmerkung (annotation)

Falls sich die Anmerkung auf einen Textbereich bezieht, ist das folgende Attribut obligat; andernfalls darf es nicht angegeben werden.

am = ... Name bzw. Nummer einer Anmerkungs-marke, die mit dem nachfolgend beschriebenen Tag <am/> definiert ist.

Wenn sich die Anmerkung auf einen Textbereich bezieht, ist ein weiteres Tag erforderlich, das den Anfang bzw. das Ende dieses Textbereichs kennzeichnet.

<am/> Anmerkungs-marke (annotation mark)

Für dieses Tag ist eines der beiden folgenden Attribute obligat:

name = ... Frei wählbarer, eindeutiger Name für die Anmerkung.

id = ... Frei wählbare, eindeutige Nummer für die Anmerkung.

Um den Anfang und das Ende des Textbereichs für den die Anmerkung gilt, festzulegen, gibt es zwei Möglichkeiten: Entweder man schreibt die Anmerkung mit dem Tag <an> an den Anfang des Bereichs und markiert das Ende des Bereichs mit dem Tag <am/> oder umgekehrt. In beiden Tags müssen die gleichen Namen bzw. die gleichen Nummern angegeben werden. Diese Zuordnung von Anmerkung und Anmerkungs-marke ist notwendig, weil sich die Textbereiche, auf die sich Anmerkungen beziehen, teilweise oder ganz überlappen können. Unabhängig davon, ob die Anmerkung in den Daten am Anfang oder am Endes des betreffenden Textbereichs steht, wird die Anmerkung auf dem rechten Rand relativ zum Bereichsende positioniert; auch die gestrichelte Linie führt in jedem Fall von der Anmerkung zum Bereichsende.

In importierten Daten kann es vorkommen, dass bei Anmerkungen, die sich auf einen Textbereich beziehen, der Anmerkungstext weder unmittelbar am Anfang noch unmittelbar am Ende des Textbereichs steht. In diesem Fall wird sowohl der Anfang als auch das Ende des Textbereichs mit dem Tag <am/> gekennzeichnet. Für diese Tags ist das folgende Attribut obligat; in allen anderen Fällen darf es nicht angegeben werden.

type = start Tag definiert den Anfang des Bereichs.

= end Tag definiert das Ende des Bereichs.

Beispiel für Anmerkungen

Zuerst eine Anmerkung, die sich nicht auf einen Textbereich bezieht. Die beiden folgenden Anmerkungen beziehen sich jeweils auf einen Textbereich, wobei sich diese beiden Textbereiche teilweise überlappen. Bei beiden Anmerkungen wird der Anmerkungstext an den Anfang des betreffenden Textbereichs geschrieben und die Anmerkungsmarke ans Ende des betreffenden Textbereichs.

Kommentar [1]: erste Anmerkung

Kommentar [2]: zweite Anmerkung

Kommentar [3]: dritte Anmerkung

Aus technischen Gründen fehlen die gestrichelten Linien von den Anmerkungen zu den betr. Stellen im Text.

```
<p>Zuerst eine Anmerkung,<an>erste Anmerkung</an> die sich
nicht auf einen Textbereich bezieht. Die beiden folgenden
Anmerkungen <an am=sonne>zweite Anmerkung</an>beziehen
sich jeweils auf einen Textbereich, wobei sich diese beiden
Textbereiche teilweise überlappen. Bei beiden Anmerkungen
<an am=mond>dritte Anmerkung</an>wird der Anmerkungstext
<am name=sonne/> an den Anfang des betreffenden Textbereichs
geschrieben und die Anmerkungsmarke<am name=mond/> ans Ende
des betreffenden Textbereichs.</p>
```

Der Anmerkungstext kann aus einem oder mehreren Absätzen bestehen. Wenn er nur aus einem Absatz besteht, können das Anfangs-Tag und das Ende-Tag für den Absatz weggelassen werden. In diesem Fall werden intern die Tags `<an-p>` und `</an-p>` ergänzt. Das Tag `<an-p>` ist vordefiniert, wobei die Attribute `font` und `size` aus dem Tag `<dstyle/>` (siehe Seite 19) bzw. dessen Voreinstellungen übernommen werden. Jedoch wird das Attribut `size` und einige andere Attribute von MS-Word innerhalb von Anmerkungen ignoriert. Das Tag `<an-p>` kann innerhalb der Definitionen auch selbst definiert werden, z. B. mit

```
<pstyle name=an-p font=Arial/>
```

Marginalien

Marginalien können wie Fußnoten in den Text eingefügt werden. Sie werden wahlweise entweder außerhalb des Satzspiegels auf dem linken oder rechten Seitenrand, innerhalb des Satzspiegels jeweils links oder rechts oder auch jeweils innen oder außen gedruckt.

Im Gegensatz zu den Anmerkungen, deren Größe automatisch angepasst wird und die automatisch untereinander angeordnet werden, muss für Marginalien die Breite und Höhe angegeben werden und ggf. müssen sie noch vertikal positioniert werden, damit sie sich nicht überlappen.

<mn> Marginalie (marginal note)

Für dieses Tag sind folgende Attribute möglich:

width	= adjust	Breite der Marginalie dem Text anpassen. [s.u.]
	= p	Marginalie p Punkt breit.
height	= adjust	Höhe der Marginalie dem Text anpassen. [s.u.]
	= p	Marginalie p Punkt hoch.
v-pos	= p	Marginalie vom oberen Satzspiegelrand aus um p Punkt nach unten positionieren.
	= ±p	Marginalie um ±p Punkt vertikal relativ zur Zeile in der sie im Text steht positionieren. [-3]
h-pos	= ±p	Marginalie um ±p Punkt horizontal relativ zum linken Satzspiegelrand positionieren. [-53]
	= left	Marginalie außerhalb des Satzspiegels auf dem linken Seitenrand. [*]
	= right	Marginalie außerhalb des Satzspiegels auf dem rechten Seitenrand.
	= left-side	Marginalie innerhalb des Satzspiegels am linken Rand.
	= right-side	Marginalie innerhalb des Satzspiegels am rechten Rand.
	= inside	Marginalie innerhalb des Satzspiegels an der Innenseite.
textflow	= outside	Marginalie innerhalb des Satzspiegels an der Außenseite.
	= normal	Text waagrecht. [*] Kann nicht mit width=adjust kombiniert werden.
	= bottom-top	Text senkrecht von unten nach oben. Kann nicht mit height=adjust kombiniert werden.
	= top-bottom	Text senkrecht von oben nach unten Kann nicht mit height=adjust kombiniert werden..
border	= none	Keinen Rahmen um die Marginalie. [*]
	= box	Rahmen um die Marginalie.
	= ellipse	Elipse als Rahmen um die Marginalie.
	= pointer	Zeiger als Rahmen um die Marginalie.
	= l-arrow	Pfeil nach links als Rahmen um die Marginalie.

	= r-arrow	Pfeil nach rechts als Rahmen um die Marginalie.
border-color	= c	Rahmen in Farbe c. [black]
border-width	= p	Rahmen p Punkt dick.
border-space	= p	Zwischen Rahmen und Text p Punkt Zwischenraum.
color	= c	Marginalie mit der Farbe c unterlegen.
opacity	= n%	Transparenz n Prozent. [0%]
option	= ignore	Text so drucken, als wäre keine Marginalie vorhanden.

Die voreingestellten Werte für die Attribute `width` und `height` sind von der Angabe zum Attribut `textflow` abhängig. Falls `textflow=normal` angegeben ist, sind `width=43` und `height=adjust` voreingestellt, andernfalls sind `width=adjust` und `height=43` voreingestellt.

Für die Höhe und Breite der Marginalien können auch ungefähre Werte angegeben werden und dann in MS-Word die Größe mit der Maus passend eingestellt werden. Außerdem können die Marginalien in MS-Word mit der Maus beliebig verschoben werden. Der Haupttext wird auch nach dem Anpassen der Größe bzw. nach dem Verschieben der Marginalie ggf. automatisch um sie herumgedruckt, falls im Tag `<mn>` nicht das Attribut `option=ignore` angegeben ist.

Der Marginalientext kann aus einem oder mehreren Absätzen bestehen. Wenn er nur aus einem Absatz besteht, können das Anfangs-Tag und das Ende-Tag für den Absatz weggelassen werden. In diesem Fall werden intern die Tags `<mn-p>` und `</mn-p>` ergänzt. Das Tag `<mn-p>` ist vordefiniert, wobei die Attribute `font`, `size` und `align` aus dem Tag `<dstyle/>` (siehe Seite 19) bzw. dessen Voreinstellungen übernommen werden. Die Schriftgröße wird jedoch um 2 Punkt reduziert, sodass die Marginalien um 2 Punkt kleiner gedruckt werden als der Text. Das Tag `<mn-p>` kann innerhalb der Definitionen auch selbst definiert werden, z. B. mit

```
<pstyle name=mn-p font=Times size=10/>
```

*Beispiel für Marginalien*linke
Seite

Erst einmal je eine Marginalie auf dem linken und dem rechten Seitenrand. Marginalien können auch an beliebigen Stellen im Text positioniert werden. Die folgende Marginalie wird zwei Zoll rechts vom linken Rand eingefügt und erhält einen schwarzen Rahmen. Der umgebende Text wird automatisch um die Marginalie herum gedruckt.

Nun links und rechts je zwei Marginalien mit je zwei Zeilen. Links wurde versehentlich beides Mal nichts zur vertikalen Position der Marginalien angegeben. Deshalb überdeckt auf der linken Seite die zweite Marginalie zum Teil die erste und von der ersten ist deshalb nur die erste Zeile zu sehen, obwohl sogar die Größe korrekt angegeben wurde.

Das ist wohl keine
Marginalie mehr.rechte
Seiteerste
links
zweite
linkserste
rechts
zweite
rechts

hier

Statt eines Rechtecks können auch andere Formen als Rahmen verwendet werden.

Zum Schluss noch eine größere Marginalie rechts.

```
<p>Erst einmal je eine Marginalie auf dem linken
<mn color=silver>linke Seite</mn> und dem rechten
<mn h-pos=right>rechte Seite</mn> Seitenrand.</p>
<p>Marginalien können auch an beliebigen Stellen im Text
positioniert werden. Die folgende Marginalie wird zwei Zoll
rechts vom linken Rand eingefügt und erhält einen schwarzen
Rahmen.<mn h-pos=+144 v-pos=-10 height=36 width=108 border=box>Das
ist wohl keine Marginalie mehr.</mn> Der umgebende Text wird
automatisch um die Marginalie herum gedruckt.</p>
<p>Nun links<mn height=30 h-pos=left>erste links</mn> und
rechts<mn h-pos=right v-pos=-13>erste rechts</mn> je zwei
Marginalien mit je zwei Zeilen. Links wurde versehentlich
beides Mal nichts zur vertikalen Position der Marginalien
<mn height=30 h-pos=left>zweite links</mn> angegeben. Deshalb
<mn h-pos=right v-pos=-1>zweite rechts</mn> überdeckt auf der
linken Seite die zweite Marginalie zum Teil die erste und
von der ersten ist deshalb nur die erste Zeile zu sehen,
obwohl sogar die Größe korrekt angegeben wurde.</p>
<p/>
<p>Statt eines Rechtecks können auch andere Formen
<mn border=pointer color=silver>hier</mn> als Rahmen
verwendet werden.</p>
<p/>
<p>Zum Schluss noch eine größere Marginalie rechts.
<mn h-pos=right height=160 textflow=top-bottom>
<font size=18>Marginalienbeispiel</font></mn>
```

Marginalienbeispiel

Weiteres Beispiel für Marginalien

Hier noch ein Text mit Marginalien, die innerhalb des Satzspiegels jeweils auf der Außenseite (d. h. auf Seiten mit ungerader Seitennummer rechts und auf Seiten mit gerader Seitennummer links) gedruckt werden. Der Text wird auch hier um die Marginalien herum gedruckt. Für die Marginalien kann auch ein entsprechend breiter Rand freigelassen werden. Da die Marginalien abhängig von der Seitennummer links oder rechts stehen, stellt sich die Frage, auf welcher Seite ein Rand freigelassen werden muss:

Der Rand muss für Seiten mit gerader Seitennummer, d. h. für linke Seiten angegeben werden. Damit dieser Rand wechselweise links bzw. rechts auf der Seite freigelassen wird, muss das Attribut 'mirror=on' angegeben werden. Wenn die Breite des Randes nicht genau der Breite der Marginalie (einschließlich des Freiraums um die Marginalie) entspricht wie in diesem Beispiel, ergibt sich noch ein Problem: Für die Zeilen, die neben der Marginalie stehen, wird der angegebene Rand nicht eingehalten. Der Text geht in jedem Fall bis zur Marginalie. Das kann vermieden werden, indem wie in diesem Beispiel die Option 'ignore' angegeben wird. Dadurch wird der Text so gedruckt, als wäre keine Marginalie vorhanden; er wird dann aber von der Marginalie verdeckt, wenn die Marginalie breiter als der Rand ist.

```
<p>Hier noch Text mit Marginalien, die innerhalb des
Satzspiegels jeweils auf der Außenseite (d. h. auf Seiten
mit ungerader Seitennummer rechts und auf Seiten mit gerader
Seitennummer links)<mn width=60 h-pos=outside>Marginalie</mn>
gedruckt werden. Der Text wird auch hier um die Marginalien
herum gedruckt. Für die Marginalien kann auch ein entsprechend
breiter Rand freigelassen werden. Da die Marginalien abhängig
von der Seitennummer links oder rechts stehen, stellt sich die
Frage, auf welcher Seite ein Rand freigelassen werden muss:</p>
```

```
<p left=84 mirror=on>Der Rand muss für Seiten mit gerader
Seitennummer, d. h. für linke Seiten angegeben werden. Damit
dieser Rand wechselweise links bzw. rechts auf der Seite
freigelassen wird, muss das Attribut 'mirror=on' angegeben
werden. Wenn die Breite des Randes nicht genau der Breite
der Marginalie<mn width=60 h-pos=outside>Marginalie</mn>
(einschließlich des Freiraums um die Marginalie) entspricht
wie in diesem Beispiel, ergibt sich noch ein Problem:
Für die Zeilen, die neben der Marginalie stehen, wird der
angegebene Rand nicht eingehalten. Der Text geht in jedem Fall
bis zur Marginalie.<mn width=60 h-pos=outside option=ignore>
Marginalie</mn> Das kann vermieden werden, indem wie in diesem
Beispiel die Option 'ignore' angegeben wird. Dadurch wird der
Text so gedruckt, als wäre keine Marginalie vorhanden; er wird
dann aber von der Marginalie verdeckt, wenn die Marginalie
breiter als der Rand ist.</p>
```

Daten ignorieren

Alle Daten, die zwischen `<!--` und `-->` stehen, werden in jedem Fall ignoriert. Außerdem können mit folgendem Tag alle untergeordneten Daten ignoriert werden.

`<ignore>` Daten ignorieren

Bei diesem Tag können beliebige Attribute angegeben sein; sie werden alle ignoriert.

Hinweis: Soll nur das Tag selbst ignoriert werden, nicht aber der Inhalt des Tags, kann das mit Hilfe des als nächstes beschriebenen Tags `<dummy>` erreicht werden.

Beispiel zum Ignorieren von Daten

Das ist ein Beispiel. Der zu ignorierende Teil kann auch beliebige Tags enthalten. Sie müssen nicht definiert sein, müssen aber korrekt geschachtelt sein.

```
<p>Das ist ein <ignore>mittelmäßiges</ignore> Beispiel. Der zu
ignorierende Teil kann auch beliebige Tags enthalten.
<ignore>Das hätte man <xy>nicht</xy> anders erwartet. </ignore>
Sie müssen nicht definiert sein, müssen aber korrekt
geschachtelt sein.</p>
```

Das vorangehende Beispiel ist sehr theoretisch. In der Praxis enthält der Text z. B. persönliche Notizen, die mit dem Tag `<notiz>` gekennzeichnet sind.

Wenn der Text ausgedruckt und dabei die persönlichen Notizen jeweils in einem eigenen Absatz in kursiver Schrift gedruckt werden sollen, kann das Tag `<notiz>` z. B. wie folgt definiert werden:

```
<pstyle name=notiz style=i/>
```

Wenn der Text als RTF-Datei ohne die persönlichen Notizen weitergegeben werden soll, kann das Tag `<notiz>` z. B. wie folgt definiert werden:

```
<alias alias=notiz tag=ignore/>
```

Mit dieser Definition hat das Tag `<notiz>` die gleichen Eigenschaften wie das vordefinierte Tag `<ignore>`, d. h. die Notizen werden beim Export in die RTF-Datei ignoriert.

Wenn der Text ausgedruckt und dabei die persönlichen Notizen wie Anmerkungen (siehe Seite 45) am rechten Rand gedruckt werden sollen, kann das Tag `<notiz>` z. B. wie folgt definiert werden:

```
<alias alias=notiz tag=an/>
```

Die Tags `<alias/>` und `<pstyle/>` sind auf Seite 92 bzw. 95 beschrieben.

Tags ignorieren

Im Gegensatz zum zuvor beschriebenen Tag `<ignore>`, bei dem der Inhalt des Tags ignoriert wird, wird beim folgenden Tag nur das Tag selbst ignoriert.

`<dummy>` Tag ignorieren

Bei diesem Tag können beliebige Attribute angegeben sein; sie werden alle ignoriert.

Beispiel zum Ignorieren von Tags

Das ist ein triviales Beispiel. Das Tag wird ignoriert, nicht aber dessen Inhalt.

```
<p>Das ist ein <dummy>triviales</dummy> Beispiel. Das Tag wird
ignoriert, nicht aber dessen Inhalt.</p>
```

Das vorangehende Beispiel ist sehr theoretisch. In der Praxis enthält der Text z. B. eine Liste mit Listeneinträgen, wobei die Listeneinträge in das Tag `<list>` eingeschlossen sind. Für den Export ist das Tag `<list>` nicht erforderlich und kann deshalb ignoriert werden. Aus diesem Grund wird im folgenden Beispiel das Tag `<list>` so definiert, dass es die gleichen Eigenschaften wie das vordefinierte Tag `<dummy>` hat:

Realistisches Beispiel zum Ignorieren von Tags

- Erster Listeneintrag
- Zweiter Listeneintrag
- Dritter Listeneintrag

```
<alias alias=list tag=dummy/>
```

```
<list>
  <li>Erster Listeneintrag</li>
  <li>Zweiter Listeneintrag</li>
  <li>Dritter Listeneintrag</li>
</list>
```

Die Tags `<alias/>` und `` sind auf Seite 92 bzw. 32 beschrieben.

Tabulatoren

Mit dem Tabulator-Tag kann jeweils von der aktuellen Position auf den nächst folgenden (weiter rechts stehenden) Tabulator positioniert werden. Die Tabulator-Positionen können für jeden Absatz mit dem Attribut `tabs` im Absatz-Tag (siehe Seite 27) definiert werden. Sind für einen Absatz keine Tabulatoren oder zu wenig Tabulatoren definiert, so wird auf den nächsten Standard-Tabulator positioniert.

Hinweis: Die Voreinstellung für den Abstand der Standard-Tabulatoren beträgt 18 Punkt. Sie kann mit dem Tag `<dstyle/>` (siehe Seite 19) festgelegt werden.

<tab/> Tabulator

Wenn für den jeweiligen Absatz keine Tabulator-Positionen definiert sind, kann in jedem Tabulator-Tag des Absatzes die Tabulator-Position mit dem Attribut `pos` festgelegt werden und ggf. mit dem Attribut `fill` das Zeichen angegeben werden, mit dem der durch den Tabulator entstehende Zwischenraum aufgefüllt werden soll:

<code>pos</code>	= p	Tabulator-Positionen auf p Punkt vom linken Satzspiegelrand setzen; nachfolgenden Text rechts davon drucken.
	= p	Tabulator-Positionen auf p Punkt vom linken Satzspiegelrand setzen; nachfolgenden Text links davon drucken.
	= p	Tabulator-Positionen auf p Punkt vom linken Satzspiegelrand setzen; nachfolgenden Text zentriert zu dieser Position drucken.
	= p,	Tabulator-Positionen auf p Punkt vom linken Satzspiegelrand setzen; Dezimalkomma (nicht Dezimalpunkt) der nachfolgenden Zahl auf dieser Position drucken.
<code>fill</code>	= .	Zwischenraum mit Punkten auffüllen.
	= ^.	Zwischenraum mit Malpunkten auffüllen.
	= -	Zwischenraum mit Minuszeichen ----- auffüllen.
	= _	Zwischenraum mit Unterstreichungsstr. ____ auffüllen.

Für den "normalen" Tabulator, bei dem der nachfolgende Text rechts ab der Tabulator-Position gedruckt wird, ist im Attribut `pos` ein senkrechter Strich vor der Zahl erforderlich!

Werden die Tabulator-Positionen nicht zusammen im Absatz-Tag, sondern jeweils in den Tabulator-Tags angegeben, so werden diese für jeden Absatz zuerst gesammelt. Die Tabulator-Tags haben dann die gleiche Wirkung wie Tabulator-Tags ohne Positionsangaben. Das Ergebnis ist deshalb in jedem Fall dasselbe, egal ob die Tabulator-Positionen zusammen im Absatz-Tag oder einzeln in den Tabulator-Tags angegeben werden.

Auch wenn die Tabulator-Position im Tabulator-Tag (z. B. `<tab pos=|75mm>`) angegeben wird, ist nicht sichergestellt, dass auch auf diese Position positioniert wird, wie das folgende Beispiel zeigt:

Beispiel für Tabulatoren

Zeile 1 vor Tab	Text nach 1. Tab	Text nach 2.Tab
Zeile 2 vor Tab	Text nach 1. Tab	Text nach 2.Tab

Text endet vor 1. Tabulatorposition	Text 1. Tab	Text 2. Tab
Dieser Text endet erst nach 1. Tabulatorposition	Text 1. Tab	Text 2. Tab

In der letzten Zeile wird mit dem ersten Tabulator auf die Position des zweiten Tabulators positioniert!

```
<p tabs=|5cm'|10cm>Zeile 1 vor Tab
<tab/>Text nach 1. Tab
<tab/>Text nach 2.Tab</p>
```

```
<p>Zeile 2 vor Tab
<tab pos=|5cm/>Text nach 1. Tab
<tab pos=|10cm/>Text nach 2.Tab</p>
```

```
<p>Text endet vor 1. Tabulatorposition
<tab pos=|75mm/>Text 1. Tab<tab pos=|110mm/>Text 2. Tab</p>
```

```
<p>Dieser Text endet erst nach 1. Tabulatorposition
<tab pos=|75mm/>Text 1. Tab<tab pos=|110mm/>Text 2. Tab</p>
```

In dieser Beschreibung werden die Attribute für die Tags in folgender Form dargestellt:

name	= wert	Beschreibung
------	--------	--------------

Bei allen Attributen stehen die Gleichheitszeichen jeweils auf der gleichen Position, die Beschreibung beginnt jeweils auf der gleichen Position und die Folgezeilen sind entsprechend eingerückt. Dafür wurde folgendes Absatz-Tag definiert:

```
<pstyle name=attr first=0 left=170 tabs=|80'|170/>
```

Damit kann die oben angegebene Zeile wie folgt geschrieben werden:

```
<attr>name<tab/>= wert<tab/>Beschreibung</attr>
```

Da MS-Word auf die mit dem Attribut `left` angegebene Position automatisch einen linksbündigen Tabulator setzt, würde auch folgende Definition genügen:

```
<pstyle name=attr first=0 left=170 tabs=|80/>
```

Bei manchen Textverarbeitungsprogrammen werden die angegebenen Tabulatorpositionen unter nicht geklärten Umständen nicht als Abstand vom linken Satzspiegelrand gewertet, sondern als Abstand von der mit dem Attribut `left` angegebenen Position. Die Angaben zu den Tabulatorpositionen müssen in diesem Fall um den zum Attribut `left` angegebenen Wert verringert werden:

```
<pstyle name=attr first=0 left=170 tabs=|-90/>
```

Zentrieren

Ob der Text eines *Absatzes* linksbündig, zentriert oder rechtsbündig gedruckt werden soll, kann im Absatz-Tag mit dem Attribut `align` (siehe Seite 26) angegeben werden.

Soll in einer *Zeile* linksbündig, zentriert und rechtsbündig kombiniert werden, so ist dies nur mit Hilfe von Tabulatoren (siehe Seite 53) möglich. Um eine Zeichenfolge in die Mitte zu drucken, muss ein Tabulator in die Mitte der Zeile gesetzt werden, um den der Text zentriert wird; um eine Zeichenfolge rechtsbündig zu drucken, muss ein Tabulator auf den rechten Rand gesetzt werden, an den der Text herangeschoben wird. Dafür können auch die folgenden vordefinierten Tags verwendet werden:

<code><center/></code>	Text auf die Zeilenmitte zentrieren
<code><right/></code>	Text an den rechten Rand schieben, Zwischenraum leer lassen
<code><fill/></code>	Text an den rechten Rand schieben, Zwischenraum punktieren

Für dieses Tags sind keine Attribute vorgesehen.

Dieser Trick mit den Tabulatoren funktioniert jedoch nur, wenn im betreffenden Absatz keine anderen Tabulatoren mit dem Attribut `tabs` (siehe Seite 27) definiert sind. Wenn das Anfangs-Tag des jeweiligen Absatzes ein eigenes, in den Definitionen definiertes Tag ist, für das mit dem Attribut `tabs` Tabulatoren definiert wurden, muss im Anfangs-Tag des jeweiligen Absatzes das Attribut `tabs=none` angegeben werden, um diese Tabulatoren für den jeweiligen Absatz zu löschen.

Beispiel für Zentrieren

Zeile 1	Zentriert mit <code><tab/></code>	Rechtsbündig mit <code><tab/></code>
Zeile 2	Zentriert mit <code><center/></code>	Rechtsbündig mit <code><right/></code>
Zeile 3	Rechtsbündig mit <code><tab/></code>
Zeile 4	Rechtsbündig mit <code><fill/></code>

```
<p>Zeile 1<tab pos=50%/>Zentriert mit ^<tab/^>
<tab pos=100%|/>Rechtsbündig mit ^<tab/^></p>
```

```
<p>Zeile 2<center/>Zentriert mit ^<center/^>
<right/>Rechtsbündig mit ^<right/^></p>
```

```
<p>Zeile 3 <tab pos=100%| fill=./> Rechtsbündig mit ^<tab/^></p>
```

```
<p>Zeile 4 <fill/> Rechtsbündig mit ^<fill/^></p>
```

Das Tag `<tab/>` ist auf Seite 53 beschrieben.

Datum, Uhrzeit, Seite

Mit den folgenden Tags kann aktuelle Information in den Text eingefügt werden. Die so eingefügten Informationen werden in der Regel vor dem Drucken automatisch aktualisiert. Sie können auch explizit aktualisiert werden, in MS-Word z. B. mit der Tastenkombination Strg+A F9.

<code><date/></code>	Datum einfügen
<code><time/></code>	Uhrzeit einfügen
<code><page/></code>	Seitennummer einfügen
<code><name/></code>	Dokumentname einfügen

Für dieses Tags sind keine Attribute vorgesehen.

Beispiel für Datum, Uhrzeit, Seite

Stand: 26. Mai 2015 09:41

Dieses Beispiel steht auf Seite 56.

In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. 56; sie enthalten jeweils den von MS-Word eingesetzten Wert.

```
<p align=right>Stand: <date/> <time/></p>
```

```
<p>Dieses Beispiel steht auf Seite <page/>.</p>  
<p/>
```

```
<p>In importierten Daten werden Anfangs- und Ende-Tags  
(sie sind im Ergebnis nicht zu sehen) verwendet,  
z._B. <page>123</page>; sie enthalten jeweils  
den von MS-Word eingesetzten Wert.</p>
```

Kopftext, Fußtext

Innerhalb einer Absatzgruppe kann mit Ausnahme der jeweils ersten Seite nur ein einheitlicher Kopftext/Fußtext gedruckt werden. Soll der Kopftext/Fußtext gewechselt werden, so muss zuvor mit dem Tag `<section>` (siehe Seite 22) eine neue Absatzgruppe begonnen werden; Ausnahme siehe "Verweis auf Absatztext" (Seite 63).

`<hdr>` Kopftext (header)

`<ftr>` Fußtext (footer)

Für beide Tags ist folgendes Attribut möglich:

<code>type</code>	<code>= first</code>	Kopftext/Fußtext für erste Seite.
	<code>= odd</code>	Kopftext/Fußtext für Seiten mit ungerader Seitennummer.
	<code>= even</code>	Kopftext/Fußtext für Seiten mit gerader Seitennummer.
	<code>= both</code>	Kopftext/Fußtext für beide Seiten. [*]

Einschränkung: Das Attribut `type=both` hat für MS-Word die gleiche Wirkung wie das Attribut `type=odd`. Der Kopftext/Fußtext muss deshalb ggf. sowohl für Seiten mit gerader Seitennummer als auch für Seiten mit ungerader Seitennummer angegeben werden, auch wenn er für beide Seiten identisch ist.

Beispiel für Fußtexte

Der sich aus dem Beispiel ergebende Fußtext steht unten auf dieser Seite.

```
<ftr type=odd><center/>- <page/> -</ftr>
<ftr type=even><center/>- <page/> -</ftr>
```

Die Tags `<center/>` und `<page/>` sind auf Seite 55 bzw. 56 beschrieben.

Der Kopftext kann aus einem oder mehreren Absätzen bestehen. Wenn er nur aus einem Absatz besteht, können das Anfangs-Tag und das Ende-Tag für den Absatz weggelassen werden. In diesem Fall werden intern die Tags `<hdr-p>` und `</hdr-p>` ergänzt. Das Tag `<hdr-p>` ist vordefiniert, wobei die Attribute `font` und `size` aus dem Tag `<dstyle/>` (siehe Seite 19) bzw. dessen Voreinstellungen übernommen werden. Das Tag `<hdr-p>` kann innerhalb der Definitionen auch selbst definiert werden, z. B.

```
<pstyle name=hdr-p size=12/>
```

Für den Fußtext gilt dasselbe, jedoch wird das Tag `<ftr-p>` verwendet.

Um eine Trennlinie nach dem Kopftext bzw. vor dem Fußtext einzufügen, muss entweder am Ende des Kopftextes bzw. Anfang des Fußtextes das Tag `<hr/>` (siehe Seite 34) eingefügt werden oder es muss im Abschnitts-Tag, das im Kopftext bzw. im Fußtext verwendet wird, (falls keines verwendet wird im Tag `<hdr-p/>` bzw. `<ftr-p/>`) das Attribut `border=bottom` bzw. `border=top` angegeben werden.

Um einen Rahmen um den Kopftext bzw. den Fußtext zu zeichnen, muss im Abschnitts-Tag, das im Kopftext bzw. im Fußtext verwendet wird, (falls keines verwendet wird im Tag `<hdr-p/>` bzw. `<ftr-p/>`) das Attribut `border=box` angegeben werden.

Lesezeichen

Mit dem folgenden Tag kann ein Lesezeichen gesetzt werden. Lesezeichen sind Stellen im Text, auf die Bezug genommen werden kann. Jedem Lesezeichen muss ein eindeutiger Name oder eine eindeutige Nummer zugeordnet werden. Ein Lesezeichen kann sich auch auf einen Textbereich beziehen; in diesem Fall muss der Anfang und das Ende des Bereichs mit einem Tag gekennzeichnet sein.

<bm/> Lesezeichen (bookmark)

Für dieses Tag ist eines der beiden ersten Attribute obligat, das dritte Attribut muss angegeben werden, wenn ein Lesezeichen für einen Bereich definiert werden soll:

<code>name</code>	= ...	Eindeutiger Name des Lesezeichen.
<code>id</code>	= ...	Eindeutige Nummer des Lesezeichen.
<code>type</code>	= <code>start</code>	Tag definiert den Anfang des Bereichs.
	= <code>end</code>	Tag definiert das Ende des Bereichs.

Ein Bereich kann nicht dadurch definiert werden, dass am Bereichsanfang ein Anfangs-Tag und am Bereichsende ein Ende-Tag geschrieben wird; es ist jeweils am Anfang und am Ende ein leeres Tag mit dem entsprechenden Attribut `type` erforderlich! Damit ist es auch möglich, Bereiche festzulegen, die sich teilweise überlappen.

Soll sich ein Lesezeichen auf einen ganzen Absatz beziehen, so kann statt eines Lesezeichen-Tags am Anfang und am Ende des Absatzes auch im Anfangs-Tag dieses Absatzes mit dem Attribut `bm` (siehe Seite 27) ein Name für das Lesezeichen angegeben werden. Entsprechendes gilt auch für Absatzgruppen, Listeneinträge sowie für Fuß- und Endnoten.

Hinweis: Auf ein Lesezeichen kann mit Verweisen (siehe Seite 60) oder Hyperlinks (siehe Seite 64) Bezug genommen werden. Dort sind auch jeweils Beispiele für die Definition von Lesezeichen und den Bezug auf Lesezeichen aufgeführt. Außerdem können Lesezeichen in den Navigationsbereich von PDF-Dateien übernommen werden (siehe nachfolgendes Kapitel).

Navigation

In MS-Word 2010 kann neben dem Text ein Navigationsbereich angezeigt werden, mit dessen Hilfe bequem zu bestimmten Stellen im Dokument gesprungen werden kann. In diesem Bereich werden die Texte derjenigen Absätze (in der Regel Überschriften) angezeigt, die mit einer entsprechenden Angabe zum Attribut `navigation` (siehe Seite 27) versehen sind.

Wird ein Dokument mit MS-Word 2010 als PDF-Datei abgespeichert, so kann unter "Optionen ..." angegeben werden, ob "Überschriften" oder "Word-Textmarken" in den Navigationsbereich der PDF-Datei übernommen werden sollen. Unter Überschriften sind in diesem Fall die Absätze gemeint, die mit einer entsprechenden Angabe zum Attribut `navigation` versehen sind, mit Word-Textmarken sind die Lesezeichen (siehe Seite 59) gemeint.

Verweise

Es kann auf Seiten, Fußnoten, Endnoten, Listeneinträge und Textteile verwiesen werden. Entsprechend wird an die Stelle des Verweises die entsprechende Seitennummer, Fußnotennummer, Endnotennummer, Nummer des Listeneintrags oder der entsprechende Textteil in das Dokument eingefügt. Das Verweisziel muss mit dem Tag `<bm>` (siehe Seite 59) als Lesezeichen definiert sein. Eine Verweisvariante ist der Hyperlink (siehe Seite 64).

Verweis auf eine Seite

`<pref/>` Verweis auf eine Seite (page reference)

Für dieses Tag ist folgendes Attribut obligat:

`bm` = ... Name oder Nummer eines Lesezeichens.

An Stelle des Tags wird vor dem Drucken automatisch die Nummer der Seite eingesetzt, auf der das Lesezeichen steht. Die Seitennummern können auch explizit eingesetzt bzw. aktualisiert werden, in MS-Word z. B. mit der Tastenkombination Strg+A F9.

In MS-Word kann mit STRG+Klick auf die eingesetzte Seitennummer zum jeweiligen Lesezeichen gesprungen werden; in einer mit MS-Word erzeugten PDF-Datei genügt dafür ein Klick (mit der linken Maustaste).

Beispiel für Seitenverweise

Am Anfang dieses Absatzes wurde ein Lesezeichen definiert, auf das in den beiden folgenden Sätzen verwiesen wird. Dieses Beispiel steht auf Seite 60.

In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. 60; sie enthalten jeweils die von MS-Word eingefügte Seitenzahl.

```
<p><bm name=seitenbeispiel/>Am Anfang dieses Absatzes wurde
ein Lesezeichen definiert, auf das in den beiden folgenden
Sätzen verweisen wird.
```

Dieses Beispiel steht auf Seite `<pref bm=seitenbeispiel/></p>`.

```
<p>In importierten Daten werden Anfangs- und Ende-Tags
(sie sind im Ergebnis nicht zu sehen) verwendet,
z. B. <pref bm=seitenbeispiel>123</pref>; sie
enthalten jeweils die von MS-Word eingesetzte Seitenzahl.</p>
```

Verweis auf Fußnote/Endnote

`<nref/>` Verweis auf eine Fußnote/Endnote (note reference)

Für dieses Tag ist folgendes Attribut obligat:

`bm` = ... Name oder Nummer eines Lesezeichens, das sich auf einen Textbereich bezieht, der eine Fußnote/Endnote enthält.

An Stelle des Tags wird vor dem Drucken automatisch die Nummer der Fußnote bzw. der Endnote eingesetzt, die innerhalb des Lesezeichenbereichs steht. Die Nummern können auch explizit eingesetzt bzw. aktualisiert werden, in MS-Word z. B. mit der Tastenkombination Strg+A F9.

In MS-Word kann mit STRG+Klick auf die eingesetzte Nummer zum jeweiligen Lesezeichen gesprungen werden; in einer mit MS-Word erzeugten PDF-Datei genügt dafür ein Klick (mit der linken Maustaste).

Beispiel für Fußnotenverweise

Am Ende dieses Satzes steht eine Fußnote, die mit einem Lesezeichen versehen ist.¹¹ Auf dieses Lesezeichen wird in den beiden folgenden Sätzen verwiesen. Die Fußnote hat die Nummer 11 und steht auf Seite 61.

In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. 11; sie enthalten jeweils die von MS-Word eingesetzte Nummer der Fuß- bzw. Endnoten.

Die sich aus dem Beispiel ergebende Fußnote steht unten auf dieser Seite.

```
<p>Am Ende dieses Satzes steht eine Fußnote, die mit einem
Lesezeichen versehen ist.<fn bm=notenbeispiel>Fußnote für
Verweisbeispiel</fn> Auf dieses Lesezeichen wird in den
beiden folgenden Sätzen verwiesen.
```

```
Die Fußnote hat die Nummer <nref bm=notenbeispiel/> und steht
auf Seite <pref bm=notenbeispiel/></p>.
```

```
<p>In importierten Daten werden Anfangs- und Ende-Tags
(sie sind im Ergebnis nicht zu sehen) verwendet,
z. B. <nref bm=notenbeispiel>123</nref>; sie
enthalten jeweils die von MS-Word eingesetzten Nummern
der Fuß- bzw. Endnoten.</p>
```

Verweis auf Listeneintrag

<lref/> Verweis auf einen Listeneintrag (list item reference)

Für dieses Tag ist folgendes Attribut obligat:

bm	= ...	Name oder Nummer eines Lesezeichens, das sich auf einen Textbereich bezieht, der einen Listeneintrag enthält.
----	-------	---------------------------------------------------------------------------------------------------------------

An Stelle des Tags wird vor dem Drucken automatisch die Nummer des Listeneintrags eingesetzt, der innerhalb des Lesezeichenbereichs steht. Die Nummern können auch explizit eingesetzt bzw. aktualisiert werden, in MS-Word z. B. mit der Tastenkombination Strg+A F9.

¹¹ Fußnote für Verweisbeispiel.

In MS-Word kann mit STRG+Klick auf die eingesetzte Nummer zum jeweiligen Lesezeichen gesprungen werden; in einer mit MS-Word erzeugten PDF-Datei genügt dafür ein Klick (mit der linken Maustaste).

Beispiel für Listenverweise

Um mit Hilfe des TUSTEP-Exports aus einer XML-Datei eine PDF-Datei zu erstellen, sind folgende Schritte erforderlich:

- 1 Umwandeln der XML- in eine TUSTEP-Datei.
- 2 Prüfen der Tags mit der Editoranweisung tp
- 3 Definieren der vorkommenden Tags.
- 4 Exportieren der Daten in eine RTF-Datei.
- 5 Öffnen der RTF-Datei mit MS-Word.
- 6 Abspeichern der Daten als PDF-Datei.

Der Schritt 3 ist natürlich nur beim ersten Mal erforderlich. Dabei kann die Anweisung tl im Editor hilfreich sein; sie zeigt eine Liste der vorkommenden Tags an.

In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. 3; sie enthalten jeweils die von MS-Word eingesetzte Nummer.

```
<lstyle name=xml2pdf label=|1 left=24/>
```

```
<p>Um mit Hilfe des TUSTEP-Exports aus einer XML-Datei eine PDF-Datei zu erstellen, sind folgende Schritte erforderlich:</p>
```

```
<xml2pdf>Umwandeln der XML- in eine TUSTEP-Datei.</xml2pdf>
<xml2pdf>Prüfen der Tags mit der Editoranweisung tp.</xml2pdf>
<xml2pdf bm=tagdef>Definieren der vorkommenden Tags.</xml2pdf>
<xml2pdf>Exportieren der Daten in eine RTF-Datei.</xml2pdf>
<xml2pdf>Öffnen der RTF-Datei mit MS-Word.</xml2pdf>
<xml2pdf>Abspeichern der Daten als PDF-Datei.</xml2pdf>
```

```
<p>Der Schritt <lref bm=tagdef/> ist natürlich nur beim ersten Mal erforderlich. Dabei kann die Anweisung tl im Editor hilfreich sein; sie zeigt eine Liste der vorkommenden Tags an.</p>
```

```
<p>In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. <lref bm=tagdef>12</lref>; sie enthalten jeweils die von MS-Word eingesetzte Nummer.</p>
```

Das Tag <lstyle> ist auf Seite 97 beschrieben.

Verweis auf Textteil

<trref/> Verweis auf einen Textteil (text reference)

Für dieses Tag ist folgendes Attribut obligat:

bm	= ...	Name oder Nummer eines Lesezeichens, das sich auf einen Textbereich bezieht.
----	-------	------------------------------------------------------------------------------

An Stelle des Tags wird vor dem Drucken automatisch der Text eingesetzt, der im Lesezeichenbereich steht. Die Texte können auch explizit eingesetzt bzw. aktualisiert werden, in MS-Word z. B. mit der Tastenkombination Strg+A F9.

In MS-Word kann mit STRG+Klick auf den eingesetzten Text zum jeweiligen Lesezeichen gesprungen werden; in einer mit MS-Word erzeugten PDF-Datei genügt dafür ein Klick (mit der linken Maustaste).

Beispiel für Textverweise

Nachmittags wird Herr/Frau nn den Vortrag über den Export von Daten halten.

Wenn im vorangehenden Satz der fehlende Name in TUSTEP oder in MS-Word eingesetzt wird, dann wird er von MS-Word automatisch in die beiden folgenden Sätze übertragen.

Nach dem Abendessen kann der Export mit eigenen Daten ausprobiert und getestet werden; Herr/Frau nn steht dabei für Fragen zur Verfügung.

In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. *Herr/Frau nn*; sie enthalten jeweils den von MS-Word eingesetzten Textteil.

```
<p style=i>Nachmittags wird <bm name=nn type=start/>Herr/Frau
nn<bm name=nn type=end/> den Vortrag über den Export von Daten
halten.</p>
```

```
<p>Wenn im vorangehenden Satz der fehlende Name in TUSTEP
oder in MS-Word eingesetzt wird, dann wird er von MS-Word
automatisch in die beiden folgenden Sätze übertragen.</p>
```

```
<p style=i>Nach dem Abendessen kann der Export mit eigenen
Daten ausprobiert und getestet werden; <tref bm=nn/> steht
dabei für Fragen zur Verfügung.</p>
```

```
<p>In importierten Daten werden Anfangs- und Ende-Tags
(sie sind im Ergebnis nicht zu sehen) verwendet,
z. B. <tref bm=nn>Herr/Frau soundso</tref>; sie
enthalten jeweils den von MS-Word eingesetzten Textteil.</p>
```

Verweis auf Absatztext

<fpref/> Verweis auf den ersten Absatz. (first paragraph reference)

<lpref/> Verweis auf den letzten Absatz. (last paragraph reference)

Für diese Tags ist folgendes Attribut obligat:

```
pstyle = ... Name eines Absatz-Tags.
```

Die beiden Tags **<fpref/>** und **<lpref/>** dürfen nur innerhalb der Tags **<hdr>** und **<ftr>** (siehe Seite 57) verwendet werden. Mit ihnen kann der Text des auf der jeweiligen Seite ersten bzw. letzten Absatzes, der den angegebenen Namen hat, in den Kopftext bzw. Fußtext eingefügt werden.

Mit Hilfe dieser beiden Tags kann der Kopftext und/oder der Fußtext geändert werden, ohne dass eine neue Absatzgruppe begonnen wird.

Hyperlinks

Ziel eines Hyperlinks kann ein Lesezeichen innerhalb desselben Dokuments, eine andere Datei oder eine Internetadresse sein. Lesezeichen müssen mit dem Tag `<bm>` (siehe Seite 59) definiert sein.

<h1> Hyperlink

Für dieses Tag ist eines der beiden ersten Attribute obligat, die restlichen sind optional:

<code>bm</code>	= ...	Name oder Nummer eines Lesezeichens.
<code>to</code>	= ...	Pfad einer Datei (einschließlich Dateiname) oder eine Internetadresse. Dem Pfad einer Datei muss <code>file:///</code> , einer Internetadresse muss <code>http://</code> bzw. <code>https://</code> vorangesetzt werden.
<code>cstyle</code>	= name	Für die Formatierung des Hyperlinks im Text die Attribute des selbst definierten Zeichenfolgen-Tags mit dem Namen <code>name</code> verwenden.
<code>style</code>	= *	Schriftform nicht ändern. [*]
	= name	Schriftform <code>name</code> , z. B. <code>i</code> für kursiv, <code>b</code> für fett. Für <code>name</code> sind alle Namen der Tags möglich, die in den Kapiteln "Schriftformen" und "Unterstreichung" (siehe Seite 39 bzw. 40) aufgeführt sind. Es können auch mehrere Namen durch Apostroph getrennt angegeben werden.
<code>bgc</code>	= c	Schrift mit der Farbe <code>c</code> unterlegen (Textmarker).
<code>fgc</code>	= c	Schrift in der Farbe <code>c</code> .
<code>ulc</code>	= c	Unterstreichung in der Farbe <code>c</code> . [black]
<code>border</code>	= none	Keinen Rahmen um die Zeichenfolge. [*]
	= box	Rahmen um die Zeichenfolge.
<code>border-color</code>	= c	Rahmen in Farbe <code>c</code> . [black]
<code>border-width</code>	= p	Rahmen <code>p</code> Punkt dick.
<code>border-space</code>	= p	Zwischen Rahmen und Text <code>p</code> Punkt Zwischenraum. [7]

Hyperlinks können in MS-Word mit STRG+Klick aktiviert werden; in einer mit MS-Word erzeugten PDF-Datei genügt dafür ein Klick (mit der linken Maustaste).

Beispiel für Hyperlinks

Vor dem nächsten Wort "Lesezeichen" wird ein Lesezeichen gesetzt, auf das weiter unten verwiesen wird. Vorher noch ein Beispiel für einen Verweis auf eine Internetadresse, unter der **Informationen zu TUSTEP** zu finden sind. Durch STRG+Anklicken des hellgrau hervorgehobenen Textes in MS-Word oder in einer damit erzeugten PDF-Datei, wird die entsprechende Internetseite aufgerufen. Nun der *Verweis auf das Wort Lesezeichen*. Durch STRG+Anklicken des kursiv hervorgehobenen Textes in MS-Word oder in einer damit erzeugten PDF-Datei springt der Cursor zum Lesezeichen oben in diesem Beispiel.

```
<p>Vor dem nächsten Wort "Lesezeichen" wird ein
<bm name=beispiel/>Lesezeichen gesetzt, auf das
weiter unten verwiesen wird. Vorher noch ein Beispiel
für einen Verweis auf eine Internetadresse, unter der
<hl to="http://www.tustep.uni-tuebingen.de" bgc=silver>
Informationen zu TUSTEP</hl> zu finden sind. Durch
STRG+Anklicken des hellgrau hervorgehobenen Textes in
MS-Word oder in einer damit erzeugten PDF-Datei, wird
die entsprechende Internetseite aufgerufen.
Nun der <hl bm=beispiel style=i>Verweis auf das Wort
Lesezeichen</hl>. Durch STRG+Anklicken des kursiv
hervorgehobenen Textes in MS-Word oder in einer damit
erzeugten PDF-Datei springt der Cursor zum Lesezeichen
oben in diesem Beispiel.</p>
```

Das Tag <bm> ist auf Seite 59 beschrieben.

Laufende Nummern

Mit dem folgenden Tag können laufende Nummern in den Text eingefügt werden. Um z. B. Tabellen und Beispiele getrennt durchzunummerieren, kann für Tabellen und Beispiele jeweils eine eigene laufende Nummer mit dem Namen 'Tabellen' bzw. 'Beispiele' verwendet werden.

<num/> Laufende Nummer

Für dieses Tag sind folgende Attribute möglich:

<code>name</code>	= ...	Name der laufenden Nummer.
<code>value</code>	= n	Laufende Nummer auf n setzen.
	= next	Laufende Nummer um 1 erhöhen. [*]
	= same	Laufende Nummer nicht erhöhen.
<code>type</code>	= decimal	Laufende Nummer als arabische Zahl in den Text einfügen. [*]
	= roman	Laufende Nummer als römische Zahl in Kleinbuchstaben in den Text einfügen.
	= ROMAN	Laufende Nummer als römische Zahl in Großbuchstaben in den Text einfügen.
	= alphabetic	Für die laufende Nummer den entsprechenden Kleinbuchstaben in den Text einfügen.
	= ALPHABETIC	Für die laufende Nummer entsprechenden Großbuchstaben in den Text einfügen.
	= hidden	Laufende Nummer nicht in den Text einfügen.
<code>bm</code>	= ...	Name für ein Lesezeichen (siehe Seite 59), das sich auf die eingefügte Nummer bezieht.

An Stelle des Tags wird vor dem Drucken automatisch die laufende Nummer eingesetzt. Die Nummern können auch explizit eingesetzt bzw. aktualisiert werden, in MS-Word z. B. mit der Tastenkombination Strg+A F9.

Mit Hilfe des Attributs `type=hidden` kann eine laufende Nummer zurückgesetzt werden, ohne dass sie in den Text eingefügt wird:

```
<num name=beispiel value=0 type=hidden/>
```

Beispiel für laufende Nummern

Beispiel 1:

Beispiel 2:

Beispiel 3:

Das Beispiel 2 zeigt ein leeres Rechteck.

In importierten Daten werden Anfangs- und Ende-Tags (sie sind im Ergebnis nicht zu sehen) verwendet, z. B. 4; sie enthalten jeweils die von MS-Word eingesetzte Nummer.

```
<p>Beispiel <num name=nr/>: <tab pos=90%| fill=./></p>
```

```
<p>Beispiel <num name=nr bm=box/>:
<color border=box><tab pos=90%|/></color></p>
```

```
<p>Beispiel <num name=nr/>: <tab pos=90%| fill=./></p>
```

```
<p>Das Beispiel <tref bm=box/> zeigt ein leeres Rechteck.</p>
```

```
<p>In importierten Daten werden Anfangs- und Ende-Tags
(sie sind im Ergebnis nicht zu sehen) verwendet,
z. B. <num name=nr>123</num>; sie enthalten
jeweils die von MS-Word eingesetzte Nummer.</p>
```

Die Tags <tab/> und <tref> sind auf Seite 53 bzw. 62 beschrieben.

Tabellen

Für Tabellen sind folgende Tags erforderlich: `<table>` und `</table>` kennzeichnen den Anfang und das Ende einer Tabelle. Mit Attributen müssen die Eigenschaften der Tabelle angegeben werden, insbesondere die Anzahl und Breite der Spalten. Die Anzahl der Zeilen ergibt sich aus der Anzahl der Tag-Paare `<tr>` und `</tr>`; sie kennzeichnen den Anfang und das Ende jeder Tabellenzeile. Mit Attributen können die Eigenschaften der Tabellenzeile angegeben werden; diese Angaben sind nur erforderlich, soweit sie von den im Tag `<table>` angegebenen abweichen. `<td>` und `</td>` kennzeichnen den Anfang und das Ende jeder Tabellenzelle. In jeder Zeile muss für jede Spalte eine Zelle angegeben werden, auch wenn die Zelle leer ist; für eine leere Zelle kann statt `<td></td>` auch das Tag `<td />` angegeben werden.

Beispiel für Tabellen

eins	zwei	drei
einerlei	zweierlei	dreierlei

```
<table cols="20%'30%'20%">
<tr><td>eins</td><td>zwei</td><td>drei</td></tr>
<tr><td>einerlei</td><td>zweierlei</td><td>dreierlei</td></tr>
</table>
```

<table> Tabelle

Für dieses Tag sind folgende Attribute möglich, das Attribut `cols` ist obligat:

<code>color</code>	<code>= c</code>	Tabelle mit der Farbe <code>c</code> unterlegen.
	<code>= c1 'c2</code>	Tabellenzeilen abwechselnd mit den Farben <code>c1</code> und <code>c2</code> unterlegen.
<code>indent</code>	<code>= p</code>	Tabelle um <code>p</code> Punkt einrücken. Dieses Attribut darf nicht zusammen mit dem Attribut <code>align</code> angegeben werden.
	<code>= left</code>	Tabelle linksbündig positionieren.
<code>align</code>	<code>= center</code>	Tabelle auf Mitte zentrieren. [*]
	<code>= right</code>	Tabelle rechtsbündig positionieren.
	<code>= none</code>	Tabelle ohne Gitter.
<code>grid</code>	<code>= p</code>	Gitter mit Linien in <code>p</code> Punkt und schwarz.
	<code>= c</code>	Gitter in Farbe <code>c</code> . [black]
	<code>= p:c</code>	Gitter mit Linien in <code>p</code> Punkt und Farbe <code>c</code> .
	<code>= p</code>	Zwischen den Tabellenspalten <code>p</code> Punkt Zwischenraum (insbesondere für Tabellen ohne Gitter). [8]

<code>row-space</code>	= <code>p</code>	Zwischen den Tabellenzeilen <code>p</code> Punkt Zwischenraum (insbesondere für Tabellen ohne Gitter). [4]
<code>header</code>	= <code>n</code>	Tabelle hat <code>n</code> Kopfzeilen, die bei jedem Seitenwechsel wiederholt werden sollen, wenn die Tabelle nicht auf eine Seite passt. [0]
<code>no-break</code>	= <code>row</code>	Innerhalb von Tabellenzeilen keine neue Seite beginnen.
<code>pstyle</code>	= ...	Name des Absatz-Tags, das verwendet werden soll, falls in Tabellenzellen keines angegeben ist. [tb-p]
	= <code>n1'n2'...</code>	In Spalte 1, 2, ... das Absatz-Tag <code><n></code> verwenden, falls in der betreffenden Tabellenzelle keines angegeben ist.
<code>cols</code>	= <code>p1'p2'...</code>	Spaltendefinitionen, siehe nachfolgenden Absatz.

Die Anzahl der Spalten wird durch die Anzahl der angegebenen Spaltendefinitionen bestimmt und muss nicht eigens angegeben werden. Eine Spaltendefinition für das Attribut `cols` besteht aus einem obligaten und aus optionalen Teilen. Obligat ist die Angabe der Spaltenbreite in `p` Punkt. Wird unmittelbar vor *oder* nach der Spaltenbreite ein senkrechter Strich angegeben, werden die Daten in der jeweiligen Spalte links- bzw. rechtsbündig positioniert; wird vor *und* nach der Spaltenbreite ein senkrechter Strich angegeben, werden die Daten im Blocksatz dargestellt; andernfalls werden sie in die Mitte der Spalte zentriert.

Vor einer solchen Angabe können jeweils noch ein oder zwei Zeichen angegeben werden, um die Positionierung bzw. Schreibrichtung des Textes in den Zellen der jeweiligen Spalte zu ändern:

- `-` Text nicht oben in den Zellen, sondern in der Mitte
- `_` Text nicht oben in den Zellen, sondern unten
- `/` Text nicht waagrecht, sondern senkrecht von unten nach oben
- `\` Text nicht waagrecht, sondern senkrecht von oben nach unten

Am Ende jeder Angabe für eine Spalte kann durch einen Doppelpunkt getrennt eine Farbe `c` angegeben werden, mit der die Spalte unterlegt werden soll.

Wird im Tag `<table>` zum Attribut `pstyle` ein Absatz-Tag angegeben, bei dessen Definition zum Attribut `align` ein anderer Wert als `none` angegeben ist, so hat dieser Wert Vorrang vor entsprechenden Angaben zum Attribut `cols` im Tag `<table>`.

`<tr>` Tabellenzeile (table row)

Für dieses Tag sind folgende Attribute möglich:

<code>color</code>	= <code>c</code>	Tabellenzeile mit der Farbe <code>c</code> unterlegen.
<code>join</code>	= ...'...'...	Tabellenzellen miteinander verbinden: <code>n-m</code> Zellen der Spalten <code>n</code> bis <code>m</code> in der aktuellen Zeile. <code>[n</code> Zellen in Spalte <code>n</code> ab der aktuellen Zeile. <code>n]</code> Zellen in Spalte <code>n</code> bis zur aktuellen Zeile.
<code>no-break</code>	= <code>within</code>	Innerhalb der Tabellenzeile keine neue Seite beginnen.
	= <code>after</code>	Nach der Tabellenzeile, d. h. zwischen dieser und der nächsten Tabellenzeile keine neue Seite beginnen.

`pstyle` = ... Name des Absatz-Tags, das verwendet werden soll, falls in Tabellenzellen keines angegeben ist.
 = `n1'n2'...` In Spalte 1, 2, ... das Absatz-Tag `<n>` verwenden, falls in der betreffenden Tabellenzelle keines angegeben ist.
`cols` = `p1'p2'...` Spaltendefinitionen, siehe nachfolgenden Absatz.

Eine Spaltendefinition `p` für das Attribut `cols` entspricht der beim gleichnamigen Attribut des Tags `<table>`, das zuvor beschrieben ist. Wird eine solche Angabe unmittelbar am Anfang mit einer Tilde `~` gekennzeichnet, so erhält die entsprechende Zelle keine Rahmen, sondern nur oben eine waagrechte Linie.

Wird im Tag `<tr>` zum Attribut `pstyle` ein Absatz-Tag angegeben, bei dessen Definition zum Attribut `align` ein anderer Wert als `none` angegeben ist, so hat dieser Wert Vorrang vor entsprechenden Angaben zum Attribut `cols` im Tag `<tr>` bzw. `<table>`.

Die Angabe von Attributen beim Tag `<tr>` ist nur erforderlich, soweit sie von den im Tag `<table>` angegebenen abweichen.

`<tc>` Tabellenzelle (table cell)

Für dieses Tag ist folgendes Attribut möglich:

`color` = `c` Tabellenzelle mit der Farbe `c` unterlegen.

Die Angabe dieses Attributs ist nur erforderlich, wenn es von dem im Tag `<tr>` bzw. im Tag `<table>` angegebenen abweicht.

Anstatt den Anfang und das Ende der Daten für die Tabellenzellen mit den Tags `<tc>` und `</tc>` zu kennzeichnen, können diese Daten auch durch das folgende Tag voneinander getrennt werden:

`<ntc/>` Nächste Tabellenzelle (next table cell)

Für dieses Tag sind keine Attribute vorgesehen.

Beispiel für Tabellen mit verbundenen Zellen

A1	B1-D1		
A2-A5	B2	C2	D2
	B3	C3	D3
	B4	C4	D4
	B5	C5	D5

```

<table cols=50'100'100'100>
<tr join=2-4>A1<ntc/>B1-D1</tr>
<tr cols=-/50'100'100'100 join=[1>A2-A5
<ntc/>B2<ntc/>C2<ntc/>D2</tr>
<tr><ntc/>B3<ntc/>C3<ntc/>D3</tr>
<tr><ntc/>B4<ntc/>C4<ntc/>D4</tr>
<tr join=1]><ntc/>B5<ntc/>C5<ntc/>D5</tr>
</table>

```

Der Tabellenzellentext kann aus einem oder mehreren Absätzen bestehen. Wenn er nur aus einem Absatz besteht, können das Anfangs-Tag und das Ende-Tag für den Absatz weggelassen werden. In diesem Fall werden intern Anfangs- und Ende-Tags mit dem im Tag `<table>` bzw. im Tag `<tr>` zum Attribut `pstyle` angegebenen Namen ergänzt. Falls mit dem Attribut `pstyle` kein Name vorgegeben ist, werden die Tags `<tb-p>` und `</tb-p>` ergänzt. Das Tag `<tb-p>` ist vordefiniert, wobei die Attribute `font` und `size` aus dem Tag `<dstyle/>` (siehe Seite 19) bzw. dessen Voreinstellungen übernommen werden. Das Attribut `align` wird auf den Wert `none` gesetzt, damit die Vorgaben in den Spaltendefinitionen für linksbündig, zentriert, rechtsbündig wirksam bleiben. Das Tag `<tb-p>` kann innerhalb der Definitionen auch selbst definiert werden, z. B.

```
<pstyle name=tb-p font=Arial align=none/>
```

Wird in Tabellenzellen ein Absatz-Tag verwendet, bei dem zum Attribut `align` ein anderer Wert als `none` angegeben ist, so hat dieser Wert Vorrang vor entsprechenden Angaben mit dem Attribut `cols` in den Tags `<tr>` bzw. `<table>`.

Um in einer Tabellenzelle z. B. die Summe aller in der gleichen Spalte darüber stehenden Zahlenwerte einzufügen, kann an Stelle der Summe folgendes Tag angegeben werden:

`<tf/>` Tabellenformel

Für dieses Tag ist folgendes Attribut obligat:

`data` = ... Funktion oder Formel zur Berechnung eines Zahlenwertes, siehe die beiden nachfolgenden Absätze.

Eine Funktion für das Attribut `data` muss die Form $f(x)$ haben. Für f darf einer der Funktionsnamen `sum`, `max`, `min` oder `average` (Summe, Maximum, Minimum oder Durchschnitt) und für x eines der folgenden Schlüsselwörter angegeben werden:

- `above` ... von allen in der gleichen Spalte darüber stehenden Zahlenwerten.
- `below` ... von allen in der gleichen Spalte darunter stehenden Zahlenwerten.
- `left` ... von allen in der gleichen Zeile links stehenden Zahlenwerten.
- `right` ... von allen in der gleichen Zeile rechts stehenden Zahlenwerten.

Eine Formel für das Attribut `data` muss eine der folgende Formen haben: $x+y$, $x-y$, $x*y$, x/y , oder $x*n\%$. Dabei muss für x und y jeweils ein Buchstabe angegeben werden, der die Spalte bezeichnet (erste Spalte = A, zweite = B, usw.), aus der in der aktuellen Zeile der Wert genommen werden soll, und für n muss eine ein- bis dreistellige Zahl angegeben werden.

Soll das Ergebnis (unabhängig von den Darstellungen der Zahlenwerte in den einbezogenen Zellen) mit einer bestimmten Anzahl von Dezimalstellen dargestellt werden, so kann die Funktion bzw. die Formel hinten mit `#n` ergänzt werden, wobei n die Anzahl der gewünschten Dezimalstellen ist.

Beispiel für Tabellen mit Summenbildung

irgend etwas	123,00 €
noch etwas	5,00 €
	128,00 €

```
<table cols="|5cm'2cm|" grid=none>
<tr>irgend etwas<ntc/>123,00 #(e)</tr>
<tr>noch etwas<ntc/>5,00 #(e)</tr>
<tr cols="|5cm'~2cm|"><ntc/><tf data=sum(above) /></tr>
</table>
```

Tabellen ermöglichen auch mehrspaltigen Parallelsatz, wie das folgende Beispiel zeigt.

Beispiel für mehrspaltigen Parallelsatz

Teil 1 links: Text, der kürzer ist als der in der letzten Spalte.	Teil 1 Mitte: irgend etwas.	Teil 1 rechts: Text, der mehr Platz in Anspruch nimmt als der Text, der in der ersten Spalte steht.
Teil 2 links: Text, der mehr Platz in Anspruch nimmt als der Text, der in der letzten Spalte steht.	Teil 2 Mitte: irgend etwas.	Teil 2 rechts: Text, der kürzer ist als der in der ersten Spalte.
Teil 3 links: Beliebiger Text.	Teil 3 Mitte: irgend etwas.	Teil 3 rechts: Beliebiger Text.

```
<table cols=|190'|90'|200 grid=none col-space=20>
<tr>Teil 1 links: Text, der kürzer ist
als der in der letzten Spalte.
<ntc/>Teil 1 Mitte: irgend etwas.
<ntc/>Teil 1 rechts: Text, der mehr Platz in Anspruch
nimmt als der Text, der in der ersten Spalte steht.</tr>
<tr>Teil 2 links: Text, der mehr Platz in Anspruch
nimmt als der Text, der in der letzten Spalte steht.
<ntc/>Teil 2 Mitte: irgend etwas.
<ntc/>Teil 2 rechts: Text, der kürzer ist
als der in der ersten Spalte.</tr>
<tr>Teil 3 links: Beliebiger Text.
<ntc/>Teil 3 Mitte: irgend etwas.
<ntc/>Teil 3 rechts: Beliebiger Text.</tr>
</table>
```

Wenn eine Tabelle ohne Gitter gedruckt wird, ist es in der Regel sinnvoll, den Freiraum zwischen den Texten zweier Spalten mit dem Attribut `col-space` zu vergrößern. Dadurch wird aber auch der Freiraum links der ersten Spalte und rechts der letzten Spalte mitvergrößert. Soll der Text der ersten Spalte am linken Satzspiegelrand beginnen und der Text der letzten Spalte bis zum rechten Satzspiegelrand reichen, so müssen die mit dem Attribut `cols` angegebenen Spaltenbreiten so vergrößert werden, dass sie zusammen so viel mehr als die Satzspiegelbreite ergeben, wie mit dem Attribut `col-space` angegeben ist. Mit dem Attribut `row-space` kann der Mindestabstand zwischen den Textteilen in den einzelnen Spalten festgelegt werden. Mit dem Attribut `no-break=row` kann verhindert werden, dass innerhalb einer Tabellenzeile eine neue Seite begonnen wird; dann stehen die einzelnen Textteile immer vollständig auf einer Seite.

Verzeichnisse

"Verzeichnis" wird in dieser Beschreibung eine Zusammenstellung von Einträgen genannt, bei der die Einträge nicht alphabetisch sortiert sind, z. B. ein Inhaltsverzeichnis oder ein Tabellenverzeichnis.

Verzeichnisse können vor dem Exportieren zusammengestellt und in die Daten eingefügt werden oder von MS-Word automatisch generiert werden. Zuerst ein Beispiel mit einem selbst erstellten Inhaltsverzeichnis.

Beispiel für eigene Inhaltsverzeichnisse

Vorwort	74
Hauptüberschrift	74
Erste Überschrift	74
Zweite Überschrift	74
Stichwortregister	74

Vorwort

...

Hauptüberschrift

...

Erste Überschrift

...

Zweite Überschrift

...

Stichwortregister

...

```

<p right=36>Vorwort <fill/> <pref bm=ü1/></p>
<p right=36>Hauptüberschrift <fill/> <pref bm=ü2/></p>
<p left=18 right=36>Erste Überschrift <fill/> <pref bm=ü3/></p>
<p left=18 right=36>Zweite Überschrift <fill/> <pref bm=ü4/></p>
<p right=36>Stichwortregister <fill/> <pref bm=ü9/></p>
<p/>
<p/>
<p no-break=title bm=ü1>Vorwort</p>
<p>. . .</p>
<p no-break=title bm=ü2>Hauptüberschrift</p>
<p>. . .</p>
<p no-break=title bm=ü3>Erste Überschrift</p>
<p>. . .</p>
<p no-break=title bm=ü4>Zweite Überschrift</p>
<p>. . .</p>
<p no-break=title bm=ü9>Stichwortregister</p>
<p>. . .</p>

```

Die Tags <fill/> und <pref> sind auf Seite 55 bzw. 60 beschrieben.

Im vorangehenden Beispiel ist in den Absatz-Tags innerhalb des Inhaltsverzeichnisses das Attribut `right=36` angegeben. Damit wird erreicht, dass überlaufende Zeilen (für den Fall, dass solche vorkämen) rechts um 36 Punkt eingezogen werden, sodass die Seitennummern freigestellt bleiben.

Für automatisch generierte Verzeichnisse gibt es drei Möglichkeiten, die Textteile auszuwählen, die in ein Verzeichnis aufgenommen werden sollen:

- Absätze (in der Regel Überschriften), deren Absatz-Tags bestimmte, frei wählbare Namen haben.
- Textteile, die mit den Tags `<toce>` oder `<toct>` gekennzeichnet sind.
- Absätze (z. B. Tabellen- und Bildunterschriften), die eine laufende Nummer mit einem Namen (siehe Seite 66) enthalten.

Verzeichnisse, bei denen die Auswahl der Einträge über den Namen einer laufenden Nummer erfolgt, werden in MS-Word "Abbildungsverzeichnisse" genannt; sie können nicht mehrstufig sein.

Die anderen Verzeichnisse können bis zu neunstufig sein. Für mehrstufige Verzeichnisse muss jeweils die Stufe (der Level) angegeben werden, auf der ein Absatz bzw. Textteil ins Verzeichnis eingefügt werden soll.

<toc/> Verzeichnis erstellen (table of contents)

Für dieses Tag sind folgende Attribute möglich:

<code>pstyle</code>	= <code>t1:n1't2:n2'...</code>	Absätze, die mit dem Absatz-Tag <code>t</code> gekennzeichnet sind, auf Level <code>n</code> (1 bis 9) ins Verzeichnis übernehmen.
<code>name</code>	= ...	Absätze, die eine laufende Nummer mit dem angegebenen Namen haben, ins Verzeichnis übernehmen.
<code>bm</code>	= ...	Name oder Nummer eines Lesezeichens, das den Textbereich bezeichnet, aus dem die Einträge für das Verzeichnis übernommen werden sollen.

Wird weder das Attribut `pstyle` noch das Attribut `name` angegeben, so werden die Textteile ins Verzeichnis übernommen, die mit dem Tag `<toce>` oder dem Tag `<toct>` gekennzeichnet sind.

Um z. B. an jedem Kapitelanfang ein zusätzliches, automatisches Inhaltsverzeichnis erstellen zu können, muss mit dem Attribut `bm` ein Lesezeichen (siehe Seite 59) angegeben werden, das den Bereich (im Beispielfall das jeweilige Kapitel) bezeichnet, aus dem die Einträge für das jeweilige Verzeichnis übernommen werden sollen. Wenn das Attribut `bm` nicht angegeben ist, werden die Einträge für das Verzeichnis aus dem gesamten Dokument übernommen.

Bevor in MS-Word Dokumente mit automatischen Verzeichnissen gedruckt oder als PDF-Datei abgespeichert werden, ist es zweckmäßig, diese zuvor z. B. mit der Tastenkombination `Strg+A F9` zwei Mal explizit zu aktualisieren, um sicherzustellen, dass die Verzeichnisse wirklich aktuell sind. Falls gefragt wird, ob "nur Seitenzahlen" oder das "gesamte Verzeichnis" aktualisiert werden soll, sollte beim ersten Durchgang (es wird nacheinander für jedes Verzeichnis gefragt) das Verzeichnis, beim zweiten Durchgang die Seitenzahlen aktualisiert werden.

Erstes Beispiel für automatische Verzeichnisse

Vorwort

...

Hauptüberschrift

...

Erste Überschrift

...

Zweite Überschrift

...

Stichwortregister

...

Inhaltsverzeichnis

Vorwort.....	76
Hauptüberschrift.....	76
Erste Überschrift.....	76
Zweite Überschrift.....	76
Stichwortregister.....	76

```
<pstyle name=Kapitel/>
```

```
<pstyle name=Überschrift/>
```

```
<Kapitel>Vorwort</Kapitel>
```

```
<p>. . .</p>
```

```
<Kapitel>Hauptüberschrift</Kapitel>
```

```
<p>. . .</p>
```

```
<Überschrift>Erste Überschrift</Überschrift>
```

```
<p>. . .</p>
```

```
<Überschrift>Zweite Überschrift</Überschrift>
```

```
<p>. . .</p>
```

```
<Kapitel>Stichwortregister</Kapitel>
```

```
<p>. . .</p>
```

```
<p style=u1>Inhaltsverzeichnis</p>
```

```
<toc pstyle=Kapitel:1'Überschrift:2/>
```

Das Tag <pstyle> ist auf Seite 95 beschrieben.

Wird im Tag `<toc/>` weder das Attribut `pstyle` noch das Attribut `name` angegeben, so wird ein Verzeichnis erstellt, in das die Textteile übernommen werden, die mit den Tags `<toce>` oder `<toct>` gekennzeichnet sind. Mit dem Tag `<toce>` gekennzeichnete Textteile bleiben im Text unsichtbar, mit dem Tag `<toct>` gekennzeichnete Textteile werden auch im Text gedruckt. Wenn das Verzeichnis mehrstufig werden soll, muss jeweils die Stufe (der Level) angegeben werden, auf der der jeweilige Textteil ins Verzeichnis eingefügt werden soll.

`<toce>` Eintrag für Verzeichnis (table of contents, entry)

`<toct>` Eintrag für Verzeichnis (table of contents, text + entry)

Für beide Tags ist folgendes Attribut möglich:

`level = n` Textteil auf Level n (1 bis 9) ins Verzeichnis einfügen. [1]

Zweites Beispiel für automatische Verzeichnisse

Vorwort

...

Hauptüberschrift

...

Erste Überschrift

...

Zweite Überschrift

...

Stichwortregister

...

Inhaltsverzeichnis

Vorwort..... 77

Hauptüberschrift..... 77

 Erste Überschrift..... 77

 Zweite Überschrift..... 77

Stichwortregister..... 77

```
<p><toct>Vorwort</toct></p>
```

```
<p>. . .</p>
```

```
<p><toct>Hauptüberschrift</toct></p>
```

```
<p>. . .</p>
```

```
<p><toct level=2>Erste Überschrift</toct></p>
```

```
<p>. . .</p>
```

```
<p><toct level=2>Zweite Überschrift</toct></p>
```

```
<p>. . .</p>
```

```
<p><toct>Stichwortregister</toct></p>
```

```
<p>. . .</p>
```

```
<p style=u1>Inhaltsverzeichnis</p>
```

```
<toc/>
```

Hinweis: Um Textteile zu kennzeichnen, die in Verzeichnisse übernommen werden sollen, können innerhalb der Definitionen auch eigene Tags definiert werden, z. B.

```
<alias alias=ivz1 tag=toct level=1/>
<alias alias=ivz2 tag=toct level=2/>
```

In diesem Beispiel werden Tags mit dem Namen 'ivz1' und 'ivz2' definiert; sie haben die gleichen Eigenschaften wie das vordefinierte Tag mit dem Namen 'toct'; mit dem Tag <ivz1> können Einträge für die Stufe 1, mit dem Tag <ivz2> Einträge für die Stufe 2 gekennzeichnet werden.

Drittes Beispiel für automatische Verzeichnisse

```
...


|     |
|-----|
| ... |
|-----|



Tabelle 1: Alte Werte


...


|     |
|-----|
| ... |
|-----|



Tabelle 2: Neue Werte


...


|     |
|-----|
| ... |
|-----|



Tabelle 3: Andere Werte


...
```

Tabellenverzeichnis

Tabelle 1: Alte Werte.....	78
Tabelle 2: Neue Werte.....	78
Tabelle 3: Andere Werte	78

```
<p>. . .</p>

<table cols=400><tr> . . . </tr></table>
<p align=center>Tabelle <num name=tabelle/>: Alte Werte</p>
<p>. . .</p>

<table cols=400><tr> . . . </tr></table>
<p align=center>Tabelle <num name=tabelle/>: Neue Werte</p>
<p>. . .</p>

<table cols=400><tr> . . . </tr></table>
<p align=center>Tabelle <num name=tabelle/>: Andere Werte</p>
<p>. . .</p>

<p style=ul>Tabellenverzeichnis</p>

<toc name=tabelle/>
```

In automatisch generierten Verzeichnissen werden die einzelnen Einträge in einem von MS-Word vorgegebenen Format gedruckt. Dieses Format kann auch selbst vorgegeben werden, indem innerhalb der Definitionen eigene Absatz-Tags definiert werden.

Verzeichnisse, deren Einträge im Tag <toc/> mit dem Attribut `name` ausgewählt werden, können nur einstufig sein. Um das Format für diese Einträge vorzugeben, kann ein Tag mit dem Namen 'toc-p' definiert werden.

Verzeichnisse, deren Einträge im Tag <toc/> nicht mit dem Attribut `name` ausgewählt werden, können mehrstufig sein. Um das Format für die Einträge der Stufe `n` vorzugeben, kann ein Tag mit dem Namen 'toc-n' definiert werden, wobei der Buchstabe `n` durch die Nummer der Stufe zu ersetzen ist. So wird z. B. mit der Definition

```
<pstyle name=toc-2 left=18 style=i right=36/>
```

erreicht, dass die Einträge auf der zweiten Stufe im Verzeichnis links um 18 Punkt eingerückt und kursiv gedruckt werden. Außerdem werden überlaufende Zeilen rechts um 36 Punkt eingezogen, sodass die Seitennummern freigestellt bleiben.

Register

"Register" wird in dieser Beschreibung eine Zusammenstellung von Einträgen genannt, bei der die Einträge alphabetisch sortiert und gleichlautende Einträge zusammengefasst sind, z. B. ein Ortsregister oder ein Personenregister.

Register können vor dem Exportieren zusammengestellt und in die Daten eingefügt werden oder von MS-Word automatisch generiert werden.

In automatisch generierte Register werden die mit den beiden nachfolgend beschriebenen Tags `<xe>` oder `<xt>` gekennzeichneten Textteile übernommen. Mit dem Tag `<xe>` gekennzeichnete Textteile bleiben im Text unsichtbar, mit dem `<xt>` gekennzeichnete Textteile werden auch im Text an der jeweiligen Stelle gedruckt; letzteres kann für Textteile verwendet werden, die unverändert ins Register übernommen werden können. Sollen mehrere Register erstellt werden, so muss jeweils noch der Name des betreffenden Registers angegeben werden.

<xe> Eintrag für Register (index entry)

<xt> Eintrag für Register (index entry + text)

Für beide Tags sind folgende Attribute möglich:

<code>name</code>	= ...	Name des Registers.
<code>ref</code>	= none	Eintrag ohne Referenz.
	= i	Eintrag mit kursiver Referenz.
	= b	Eintrag mit fetter Referenz.
<code>bm</code>	= ...	Name oder Nummer eines Lesezeichens.

Wird mit dem Attribut `bm` ein Lesezeichen (siehe Seite 59) angegeben, so wird als Referenz ein Seitenbereich in der Form `n-m` eingefügt, wobei `n` die erste und `m` die letzte Seite ist, auf die sich das Lesezeichen bezieht.

Beispiel für einen Eintrag ohne Referenz:

```
<xe ref=none>irgend etwas <i>siehe</i> irgend wo</xe>
```

Mit folgendem Tag kann ein Eintrag für ein mehrstufiges Register unterteilt werden:

<xs/> Untereintrag für Register (index subentry)

Für dieses Tag sind keine Attribute vorgesehen.

Beispiel für einen Eintrag mit einem Untereintrag:

```
<xe>Haupteintrag<xs/>Untereintrag</xe>
```

Als Referenz erhalten die Einträge die jeweilige Seitennummern. Mit Hilfe des folgenden Tags kann einem Eintrag jedoch ein beliebiger Text als Referenz zugewiesen werden.

<xr/> Referenz für Registereintrag (index reference)

Für dieses Tag sind keine Attribute vorgesehen.

Dieses Tag muss nach dem Eintrag, aber noch vor dem Ende-Tag `</xe>` bzw. `</xt>` angegeben werden. Als Referenz gilt die zwischen dem Tag `<xr/>` und dem Ende-Tag stehende Zeichenfolge.

Beispiel für einen Eintrag mit vorgegebener Referenz:

```
<xe>irgend etwas<xr/>referenz</xe>
```

Das Tag `<xr/>` darf nicht bei Einträgen verwendet werden, für die im Tag `<xe>` bzw. `<xt>` mit dem Attribut `bm` schon eine Referenz festgelegt wurde oder bei denen mit dem Attribut `ref=none` angegeben wurde, dass der Eintrag keine Referenz erhalten soll.

<index/> Register erstellen

Für dieses Tag sind folgende Attribute möglich:

<code>name</code>	= ...	Name des Registers.
<code>add</code>	= letter	Bei Buchstabenwechsel Extra-Buchstabe einfügen. [*]
	= space	Bei Buchstabenwechsel Leerzeile einfügen.
	= nothing	Bei Buchstabenwechsel nichts einfügen.
<code>align</code>	= none	Referenzen direkt hinter dem Eintrag. [*]
	= right	Referenzen rechtsbündig, Zwischenraum auspunktet.
<code>cols</code>	= n	Register n-spaltig (n = 1 bis 4) erstellen. [2]

Um im folgenden Beispiel Schreibearbeit zu sparen, werden drei eigene Tags definiert: die Tags `<OR>` und `<or>`, um Registereinträge für das Ortsregister zu kennzeichnen, und das Tag `<PR>`, um Einträge für das Personenregister zu kennzeichnen:

```
<alias alias=or tag=xt name=ort/>
```

```
<alias alias=OR tag=xe name=ort/>
```

```
<alias alias=PR tag=xe name=person/>
```

Mit dem Tag `<or>` können Textteile gekennzeichnet werden, die unverändert auch als Registereintrag ins Ortsregister übernommen werden sollen. Mit den Tags `<OR>` und `<PR>` können Registereinträge gekennzeichnet werden, die ins Ortsregister bzw. Personenregister übernommen werden sollen. Als Voreinstellung für das Attribut 'name' wird jeweils der frei wählbare Name des Registers angegeben. Anstatt z. B.

```
<xt name=ort>Ortsangabe</xt>
```

für einen Ortsregistereintrag zu schreiben, kann mit obiger Definition kurz

```
<or>Ortsangabe</or>
```

geschrieben werden.

Beispiel für Register

Der alleinige Zweck dieses Textes ist ein Personen- und ein Ortsregister zu erstellen. Dazu reisen Peter Müller und seine Frau Sabine von Berlin über München nach Bozen. Dort treffen sie Karl Schmid und seine Freundin Inge. Tags darauf gehen sie über die Oswaldpromenade zum Schloss Runkelstein, von dort zurück zum Bozner Obstmarkt und dann ins Archäologiemuseum, um den 5000 Jahre alten "Ötzi" zu sehen. Am nächsten Tag fahren sie nach Brixen und machen noch einen Abstecher nach Kloster Neustift. Damit sich mehr Registereinträge ergeben, reisen Josef Maier aus Trier und Fritz Sailer aus Zürich an. Tags darauf fahren sie gemeinsam nach Meran, gehen durch die Lauben und über die Kurpromenade. Nachmittags gehen sie auf dem Sissiweg zum botanischen Garten von Schloss Trauttmansdorff.

Personenregister

H		Müller, Sabine	81
Häberle, Inge	81	S	
M		Sailer, Fritz	81
Maier, Josef	81	Schmid, Karl	81
Müller, Peter	81		

Ortsregister

Berlin	81	Meran	81
Bozen	81	Kurpromenade	81
Archäologiemuseum	81	Lauben	81
Obstmarkt	81	Trauttmansdorff	81
Oswaldpromenade	81	München	81
Runkelstein	81	Trier	81
Brixen	81	Zürich	81

```
<alias alias=or tag=xt index=ort/>
<alias alias=OR tag=xe index=ort/>
<alias alias=PR tag=xe index=person/>
```

```
<p>Der alleinige Zweck dieses Textes ist ein Personen- und ein
Ortsregister zu erstellen. Dazu reisen Peter Müller<PR>Müller,
Peter</PR> und seine Frau Sabine<PR>Müller, Sabine</PR> von
<or>Berlin</or> über <or>München</or> nach <or>Bozen</or>.
Dort treffen sie Karl Schmid<PR>Schmid, Karl</PR> und seine
Freundin Inge<PR>Häberle, Inge</PR>. Tags darauf gehen sie
über die Oswaldpromenade<OR>Bozen<xs/>Oswaldpromenade</OR> zum
Schloss Runkelstein<OR>Bozen<xs/>Runkelstein</OR>, von dort
zurück zum Obstmarkt<OR>Bozen<xs/>Obstmarkt</OR> und dann ins
Archäologiemuseum<OR>Bozen<xs/>Archäologiemuseum</OR>, um den
5000 Jahre alten "Ötzi" zu sehen. Am nächsten Tag fahren sie
nach <or>Brixen</or> und machen noch einen Abstecher nach
Kloster Neustift. Damit sich mehr Registereinträge ergeben,
reisen Josef Maier<PR>Maier, Josef</PR> aus <or>Trier</or>
und Fritz Sailer<PR>Sailer, Fritz</PR> aus <or>Zürich</or>
an. Tags darauf fahren sie gemeinsam nach <or>Meran</or>,
gehen durch die Lauben<OR>Meran<xs/>Lauben</OR> und über die
Kurpromenade<OR>Meran<xs/>Kurpromenade</OR>. Nachmittags
gehen sie auf dem Sissiweg zum botanischen Garten von Schloss
Trauttmansdorff<OR>Meran<xs/>Trauttmansdorff</OR>.</p>
```

```
<p style=ul>Personenregister</p>
<index name=pers cols=2/>
```

```
<p style=ul>Ortsregister</p>
<index name=ort cols=2 add=nothing align=right/>
```

Das Tag <alias> ist auf Seite 92 beschrieben.

In den Registern werden die einzelnen Einträge in einem von MS-Word vorgegebenen Format gedruckt. Um das Format vorzugeben, kann für die Einträge jeder Stufe der Register innerhalb der Definitionen ein eigenes Absatz-Tag definiert werden. Für diese Tags müssen die Namen 'index-n' verwendet werden, wobei der Buchstabe n durch Nummer der Stufe zu ersetzen ist. So wird z. B. mit der Definition

```
<pstyle name=index-2 left=18 style=i/>
```

erreicht, dass die Einträge auf der zweiten Stufe in Registern um 18 Punkt links eingerückt und kursiv gedruckt werden. Um das Format für den Extra-Buchstaben bzw. die Leerzeile bei Buchstabenwechsel vorzugeben, kann ein entsprechendes Absatz-Tag mit dem Namen 'index-0' definiert werden.

Bilder

Mit dem folgenden Tag können Bilder, die im JPEG- oder PNG-Format vorliegen, in das Dokument eingefügt werden.

<picture/> Bild einfügen

Für dieses Tag sind folgende Attribute möglich, wobei das Attribut `file` obligat ist:

<code>path</code>	= <code>pfad</code>	Pfad (ohne Dateiname) zu der Datei, die das Bild enthält. Es kann ein absoluter Pfad oder ein relativer (mit Punkt beginnender) Pfad angegeben werden. Ein relative Pfad bezieht sich auf das Verzeichnis, in dem die QUELL-Datei steht. [.]
<code>file</code>	= <code>datei</code>	Name der Datei, die das Bild enthält.
<code>scale</code>	= <code>n%</code>	Bild auf <code>n</code> Prozent skalieren. [100%]
<code>width-scale</code>	= <code>n%</code>	Bild in der Breite auf <code>n</code> Prozent skalieren. [100%] Dieses Attribut darf nicht zusammen mit dem Attribut <code>scale</code> angegeben werden.
<code>height-scale</code>	= <code>n%</code>	Bild in der Höhe auf <code>n</code> Prozent skalieren. [100%] Dieses Attribut darf nicht zusammen mit dem Attribut <code>scale</code> angegeben werden.
<code>width</code>	= <code>p</code>	Bild <code>p</code> Punkt breit.
<code>height</code>	= <code>p</code>	Bild <code>p</code> Punkt hoch.
<code>v-pos</code>	= <code>p</code>	Bild vom oberen Satzspiegelrand aus um <code>p</code> Punkt nach unten positionieren.
	= <code>±p</code>	Bild vertikal um <code>±p</code> Punkt relativ zu der Zeile, in der das Tag <code><picture/></code> im Text steht, positionieren. [+0]
<code>h-pos</code>	= <code>±p</code>	Bild horizontal um <code>±p</code> Punkt relativ zum linken Satzspiegelrand positionieren. [0]
	= <code>left</code>	Bild am linken Satzspiegelrand positionieren. [*]
	= <code>right</code>	Bild am rechten Satzspiegelrand positionieren.
<code>border</code>	= <code>none</code>	Keinen Rahmen um das Bild. [*]
	= <code>box</code>	Rahmen um das Bild.
<code>border-color</code>	= <code>c</code>	Rahmen in Farbe <code>c</code> . [black]
<code>border-width</code>	= <code>p</code>	Rahmen <code>p</code> Punkt dick.
<code>border-space</code>	= <code>p</code>	Zwischen Rahmen und Bild <code>p</code> Punkt Zwischenraum. [0]
<code>text</code>	= <code>around</code>	Text um das Bild herum drucken.
	= <code>above</code>	Text über das Bild drucken.
	= <code>below</code>	Text hinter das Bild drucken.

Es gibt zwei Methoden, ein Bild in den Text einzufügen.

Bei der ersten Methode wird das Bild genau an der Stelle in den laufenden Text eingefügt, an der das Tag `<picture/>` im Text steht, so wie ein übergroßes Sonderzeichen eingefügt würde. Das Bild kann natürlich auch alleine in einer Zeile bzw. in einem Absatz stehen. Im letzteren Fall kann die Größe des Freiraums um das Bild mit Attributen im Anfangs-Tag des Absatzes vorgegeben werden. Außer den Attributen `path` und `file` darf bei dieser Methode nur noch ein Skalierungsfaktor mit dem Attribut `scale` angegeben werden. Ist außer den Attributen `path` und `file` kein weiteres Attribut angegeben, dann wird so verfahren, als wäre `scale=100%` angegeben.

Erstes Beispiel zum Einfügen von Bildern



Bildunterschrift



Bilderserie

```
<p align=center>
<picture file=Blume.jpg scale=50%/>
<br/>
<br/>Bildunterschrift
</p>
```

```
<p/>
```

```
<p align=center>
<picture file=Blume.jpg scale=20%/><enspace/>
<picture file=Blume.jpg scale=20%/><enspace/>
<picture file=Blume.jpg scale=20%/>
<br/>
<br/>Bilderserie
</p>
```

Bei der zweiten Methode, ein Bild in den Text einzufügen, muss angegeben werden, wo das Bild positioniert und wie groß das Bild gedruckt werden soll. Die Positionsangabe gilt für die linke obere Ecke des Bildes. Sie bezieht sich auf den Anfang der Zeile im Ausdruck, in der das Tag `<picture/>` steht. Falls das Attribut `h-pos=right` angegeben ist, wird die horizontale Position berechnet, indem von der Satzspiegelbreite die mit dem Attribut `width` angegebene Bildbreite abgezogen wird. Ist weder das Attribut `width` noch das Attribut `height` angegeben, wird das Bild auf Satzspiegelbreite skaliert. Ist eines der beiden Attribute angegeben, so wird der Wert des anderen so gewählt, dass das Seitenverhältnis des Bildes erhalten bleibt. Sind beide Attribute angegeben, wird das Bild ggf. so gedehnt/gestaucht, dass es in der angegebene Größe dargestellt werden kann. Der Text wird um das Bild herum gedruckt. Im Gegensatz zur ersten Methode wird auch der freie Raum links und rechts des Bildes mit Text gefüllt.

Zweites Beispiel zum Einfügen von Bildern



Dieser Text wird um das Bild herum gedruckt. Das geht alles ganz gut, wenn die Angabe zum Attribut `v-pos` so gewählt wird, dass das Bild nicht oder zumindest nicht weit über den oberen oder unteren Satzspiegelrand hinausragt. Andernfalls wird das Ergebnis oft unbefriedigend und es hilft nur, entweder den Umbruch zu ändern oder das Bild an einer anderen Stelle einzufügen. Auf jeden Fall sollten Bilder, die mit der zweiten Methode in den Text eingefügt werden, in MS-Word kontrolliert und ggf. mit der Maus an eine andere Stelle verschoben werden. Es ist beobachtet worden, dass am unteren Seitenrand Text vom Bild verdeckt war und erst wieder zum Vorschein kam, als das Bild mit der Maus verschoben wurde.

`<p><picture file=Blume.jpg width=150 height=100/>`

Dieser Text wird um das Bild herum gedruckt. Das geht alles ganz gut, wenn die Angabe zum Attribut v-pos so gewählt wird, dass das Bild nicht oder zumindest nicht weit über den oberen oder unteren Satzspiegelrand hinausragt. Andernfalls wird das Ergebnis oft unbefriedigend und es hilft nur, entweder den Umbruch zu ändern oder das Bild an einer anderen Stelle einzufügen. Auf jeden Fall sollten Bilder, die mit der zweiten Methode in den Text eingefügt werden, in MS-Word kontrolliert und ggf. mit der Maus an eine andere Stelle verschoben werden. Es ist beobachtet worden, dass am unteren Seitenrand Text vom Bild verdeckt war und erst wieder zum Vorschein kam, als das Bild mit der Maus verschoben wurde.`</p>`

Bastelecke

Mit dem folgenden Tag kann die Schreibposition so verändert werden, dass Zeichen ggf. auch übereinander geschrieben werden.

<move/> Schreibposition verschieben

Für dieses Tag sind folgende Attribute möglich:

v-pos	= p	Anfang des aktuellen Absatzes vom oberen Satzspiegelrand aus um mindestens p Punkt nach unten setzen.
	= -p	Schreibposition um p Punkt nach oben verschieben.
	= +p	Schreibposition um p Punkt nach unten verschieben.
h-pos	= p	Schreibposition vom linken Satzspiegelrand aus p Punkt nach rechts setzen.
	= -p	Schreibposition um p Punkt nach links verschieben.
	= +p	Schreibposition um p Punkt nach rechts verschieben.

Bei beiden Attributen wird der angegebene Wert auf ganze Punkt gerundet.

Mit dem folgenden Tag kann ein "Feld" in das Dokument eingefügt werden.

<field/> Feld einfügen

Für dieses Tag ist das erste Attribut obligat, das zweite ist optional:

inst	= ...	Feldanweisung an MS-Word.
rslt	= ...	Feldinhalt, er wird angezeigt bis das Feld aktualisiert wird.

Soll mit dem Attribut `rslt` ein Backslash an MS-Word übergeben werden, so muß dieser mit `^\` codiert werden.

Definitionen

Zum Exportieren von Daten können eigene Tags definiert und auch vordefinierte umdefiniert werden. Die Definitionen müssen entweder in der zur Spezifikation DEFINITIONEN angegebenen Datei oder am Anfang der QUELL-Datei vor dem Text stehen.

Wenn die Definitionen in der QUELL-Datei stehen, darf außer den Tags `<document>` und `<head>` nichts davor stehen (siehe Kapitel "Datenstruktur" Seite 14).

Zu den Definitionen zählt auch das Tag `<dstyle/>`. Es ist im Kapitel "Voreinstellungen" auf Seite 19 beschrieben.

Innerhalb der Definitionen dürfen keinerlei Daten außer dem Tag `<dstyle/>`, den in diesem Kapitel beschriebenen Tags und in `<!--` und `-->` eingeschlossene Kommentare vorkommen. Das Tag `<macro>` zum Definieren eines Makros kann jedoch Text, vordefinierte und selbst definierte Tags enthalten.

Wenn die Definitionen in einer separaten Datei stehen, können sie mit den Tags `<export-defs>`, `<value-defs>`, `<alias-defs>`, `<sstyle-defs>`, `<pstyle-defs>`, `<lstyle-defs>`, `<cstyle-defs>`, `<macro-defs>` und `<code-defs>` strukturiert werden.

Mit den in diesem Kapitel nachfolgend beschriebenen Tags kann folgendes definiert werden:

- Namen für Attributwerte
- Alias-Namen für Tags
- Tags für Absatzgruppen
- Tags für Absätze
- Tags für Listen
- Tags für Zeichenfolgen
- Makros
- Code-Tabellen

Die selbst definierten Tags für Absätze, Listen und Zeichenfolgen werden von MS-Word als Formatvorlagen übernommen.

Mit Ausnahme des Tags zum Definieren von Attributwerten können bei allen in diesem Kapitel beschriebenen Tags folgende zwei Attribute angegeben werden:

`ignore` = `name1'name2'...` Namen der Attribute, die ignoriert werden sollen, wenn sie im neuen Tag vorkommen.

`rename` = `ist1:soll1'ist2:soll2'...` Namen von Attributen, wobei `ist` jeweils der Name ist, der in den Daten verwendet wird, und `soll` jeweils der Name des Attributs ist, dessen Funktion das Attribut mit dem Namen `ist` haben soll.

Wie diese beiden Attribute verwendet werden können, soll ein Beispiel zeigen.

In den Daten wurde folgendes Tag verwendet, um einen Absatz zu kennzeichnen:

```
<beitrag border=box bearb=Maier>...</beitrag>
```

Mit dem Attribut `<border>` wird eine Rahmen um den Absatz gezeichnet, mit dem Attribut `bearb` ist der Bearbeiter angegeben; letzteres hat keine Bedeutung für den Ausdruck und soll ignoriert werden. Das Tag `beitrag` kann so definiert werden:

```
<pstyle name=beitrag ignore=bearb/>
```

Wenn darüber hinaus für linken und rechten Einzug die Attribute `le` und `re` statt der Attribute `left` und `right` verwendet werden, kann das Tag so definiert werden:

```
<pstyle name=beitrag ignore=bearb rename=le:left're:right/>
```

Namen für Attributwerte

<values/> Definition eines Namens für einen Attributwert

Für dieses Tag sind die Namen und Werte der Attribute frei wählbar; jedoch darf der Name kein Schlüsselwort sein, das als Attributwert vorgesehen ist (z. B. `left`), da sonst nicht erkennbar wäre, ob mit dem Attributwert (z. B. bei `align=left`) das vordefinierte Schlüsselwort oder der selbst definierte Name eines Attributwertes gemeint ist.

`name` = `wert` Name `name` für den Wert `wert` definieren.

Nachdem mit diesem Tag ein Name für einen Attributwert definiert wurde, kann in Attributen statt eines Attributwertes der Name des Wertes angegeben werden.

In manchen Umgebungen werden solche selbst definierte Namen auch Variablen genannt und müssen mit einem Dollarzeichen unmittelbar vor dem Namen gekennzeichnet werden. Diese Kennzeichnung ist möglich; das Dollarzeichen muss in diesem Fall auch bei der Definition im Tag `<values>` angegeben werden, so als wäre es Bestandteil des Namens.

Beispiel für selbst definierte Namen als Attributwerte

Kleine Schrift `mittlere Größe hellgrau unterlegt` wieder klein

```
<values hellgrau=E0E0E0 dunkelgrau=C0C0C0/>
<values klein=10 mittel=12 groß=16/>
```

```
<p size=klein>Kleine Schrift
<font size=mittel bgc=hellgrau>mittlere Größe
hellgrau unterlegt</font>
wieder klein</p>
```

Das Tag `` ist auf Seite 37 beschrieben.

Anmerkung: Für diese Beschreibung wurden mit den beiden am Anfang des vorangehenden Beispiels stehenden Tags, die im Ergebnis natürlich nicht zu sehen sind, Namen für Farben und die verwendeten Schriftgrößen definiert. Diese Namen wurden z. B. auch schon bei der Definition von Absatz- und Zeichenfolgen-Tags für die Schriftgrößen verwendet. Auf diese Weise müssen die verwendeten Schriftgrößen ggf. nur an einer einzigen Stelle geändert werden.

Alias-Namen für Tags

Für die im Kapitel "Datenstruktur" (siehe Seite 14) beschriebenen Tags `<document>`, `<head>` und `<body>`, das im Kapitel "Voreinstellungen" (siehe Seite 19) beschriebene Tag `<dstyle/>` sowie die in diesem Kapitel beschriebenen Tags `<sstyle/>`, `<pstyle/>`, `<lstyle/>`, `<cstyle/>`, `<values/>` und `<alias/>` können keine Alias-Namen definiert werden.

Für alle übrigen vordefinierten Tags können Alias-Namen definiert werden. Für selbst definierte Tags können ebenfalls Alias-Namen definiert werden; diese Tags müssen jedoch jeweils vor dem Alias-Namen definiert werden. Für Alias-Namen können keine Alias-Namen definiert werden.

`<alias/>` Definition eines Alias-Namens für Tags

Für dieses Tag sind die beiden folgenden Attribute obligat:

<code>alias</code>	= ...	Name des neu zu definierenden Tags.
<code>tag</code>	= <code>name</code>	Name des vordefinierten Tags, dessen Funktion das neue Tag haben soll.
	= <code>~name</code>	Name des selbst definierten Tags, dessen Funktion das neue Tag haben soll.

Im Gegensatz zu allen anderen Tag-Definitionen muss bei der Definition eines Alias-Namens der Name des neuen Tags nicht mit dem Attribut 'name', sondern mit dem Attribut 'alias' angegeben werden. Das Attribut 'name' kann dafür nicht verwendet werden, weil das Tag für das der Alias-Name definiert wird, möglicherweise ein Attribut 'name' hat, und sich somit eine Doppelbelegung ergäbe, wie das folgende Beispiel zeigt:

```
<alias alias=PR tag=x name=person/>
```

Damit wird ein Tag `<PR>` definiert, das statt des folgenden Tags verwendet werden kann:

```
<x name=person/>
```

Ein mit `<alias/>` definiertes Tag kann in gleicher Weise verwendet werden, wie das entsprechend vordefinierte bzw. zuvor selbst definierte Tag. Es sind automatisch auch alle Attribute möglich, die für das jeweilige Tag vorgesehen sind.

Sollen für das neue Tag bestimmte Attributwerte voreingestellt werden, so können diese bei der Definition im Tag `<alias/>` angegeben werden.

In Alias-Tags für die Tags `<pref/>`, `<nref/>`, `<lref/>` und `<tref/>`, die im Kapitel "Verweise" (siehe Seite 60) beschrieben sind, sowie für das Tag `<num/>`, das im Kapitel "Laufende Nummern" (siehe Seite 66) beschrieben ist, sind außerdem noch die beiden folgenden Attribute möglich:

<code>start</code>	= ...	Zeichenfolge, die unmittelbar vor der Referenz bzw. Nummer in den Text eingefügt wird.
<code>end</code>	= ...	Zeichenfolge, die unmittelbar nach der Referenz bzw. Nummer in den Text eingefügt wird.

Zusätzlich sind im Tag `<alias/>` die beiden folgenden Attribute möglich:

`ignore` = `name1'name2'...` Namen der Attribute, die ignoriert werden sollen, wenn sie im neuen Tag vorkommen.

`rename` = `ist1:soll1'ist2:soll2'...` Namen von Attributen, wobei `ist` jeweils der Name ist, der in den Daten verwendet wird, und `soll` jeweils der Name des Attributs ist, dessen Funktion das Attribut mit dem Namen `ist` haben soll.

Beispiel für Alias-Namen

Für diese Beschreibung wurde mit dem am Anfang dieses Beispiels stehenden Tag, das im Ergebnis natürlich nicht zu sehen ist, ein Alias-Name für das Tag `<pref/>` definiert. Hier wurde ein Lesezeichen definiert, auf das nachher verwiesen wird.

...

In dem Beispiel zum Definieren von Alias-Namen für Tags (siehe Seite 93) wird ein Tag definiert, mit dem ein Seitenverweis der Form "(siehe Seite ...)" in den Text eingefügt werden kann.

```
<alias alias=siehe tag=pref start="(siehe Seite " end=")"/>
```

```
<p>Für diese Beschreibung wurde mit dem am Anfang dieses
Beispiels stehenden Tag, das im Ergebnis natürlich nicht zu
sehen ist, ein Alias-Name für das Tag <pref/> definiert.
<bm name=aliasbeispiel/>Hier wurde ein Lesezeichen definiert,
auf das nachher verwiesen wird.</p>
```

```
<p>. . .</p>
```

```
<p>In dem Beispiel zum Definieren von Alias-Namen für Tags
<siehe bm=aliasbeispiel/> wird ein Tag definiert, mit dem
ein Seitenverweis der Form "(siehe Seite ...)" in den Text
eingefügt werden kann.</p>
```

Die Tags `<pref/>` und `<bm/>` sind auf Seite 60 bzw. 59 beschrieben.

Mit einer Alias-Definition können auch für ein vordefiniertes Tag Voreinstellungen für Attributwerte festgelegt werden. So kann z. B. mit der Definition

```
<alias alias=table tag=table cols=|100'|200'|100>
```

die Spaltenbreite der Tabellen für die Fälle festgelegt werden, in denen im Tag `<table>` das Attribut 'cols' nicht angegeben wird.

Tags für Absatzgruppen

<sstyle/> Definition eines Absatzgruppen-Tags (section style)

Für dieses Tag ist folgendes Attribut obligat:

name = ... Name des neuen Tags.

Ein mit `<sstyle/>` definiertes Tag kann in gleicher Weise verwendet werden, wie das vordefinierte Tag `<section/>` (siehe Seite 22). Es sind automatisch auch alle Attribute möglich, die für das Tag `<section>` vorgesehen sind.

Sollen für das neue Tag bestimmte Attributwerte voreingestellt werden, so können diese im Tag `<sstyle/>` angegeben werden. Für das Attribut `bm` kann jedoch keine Voreinstellung angegeben werden.

Sollen für das vordefinierte Tag `<section>` andere Attributwerte voreingestellt werden, kann es mit dem Tag `<sstyle/>` umdefiniert werden.

Zusätzlich sind im Tag `<sstyle/>` die beiden folgenden Attribute möglich:

ignore = name1'name2'... Namen der Attribute, die ignoriert werden sollen, wenn sie im neuen Tag vorkommen.

rename = ist1:soll1'ist2:soll2'... Namen von Attributen, wobei `ist` jeweils der Name ist, der in den Daten verwendet wird, und `soll` jeweils der Name des Attributs ist, dessen Funktion das Attribut mit dem Namen `ist` haben soll.

Tags für Absätze

<pstyle/> Definition eines Absatz-Tags (paragraph style)

Für dieses Tag ist folgendes Attribut obligat:

name = ... Name des neuen Tags.

Ein mit `<pstyle/>` definiertes Tag kann in gleicher Weise verwendet werden, wie das vordefinierte Tag `<p>` (siehe Seite 25). Es sind automatisch auch alle Attribute möglich, die für das Tag `<p>` vorgesehen sind.

Sollen für das neue Tag bestimmte Attributwerte voreingestellt werden, so können diese im Tag `<pstyle/>` angegeben werden. Für das Attribut `bm` kann jedoch keine Voreinstellung angegeben werden.

Sollen für das vordefinierte Tag `<p>` andere Attributwerte voreingestellt werden, kann es mit dem Tag `<pstyle/>` undefiniert werden.

Zusätzlich sind im Tag `<pstyle/>` die folgenden Attribute möglich:

<code>next-first</code>	= <code>first</code>	Das Attribut <code>first</code> im nächsten Absatz berücksichtigen; d. h. erste Zeile entsprechend dem Attribut <code>first</code> einrücken. [*]
	= <code>left</code>	Das Attribut <code>first</code> im nächsten Absatz nur berücksichtigen, wenn es explizit im Anfangs-Tag angegeben ist, oder wenn damit ein kleinerer Wert angegeben ist, als mit dem Attribut <code>left</code> ; andernfalls erste Zeile entsprechend dem Attribut <code>left</code> wie Folgezeilen einrücken.
<code>start</code>	= ...	Zeichenfolge, die unmittelbar nach dem Anfangs-Tag in den Text eingefügt wird.
<code>end</code>	= ...	Zeichenfolge, die unmittelbar vor dem Ende-Tag in den Text eingefügt wird.
<code>data</code>	= ...	Zeichenfolge, die an Stelle des leeren Tags in den Text eingefügt wird.
<code>ignore</code>	= <code>name1'name2'...</code>	Namen der Attribute, die ignoriert werden sollen, wenn sie im neuen Tag vorkommen.
<code>rename</code>	= <code>ist1:soll1'ist2:soll2'...</code>	Namen von Attributen, wobei <code>ist</code> jeweils der Name ist, der in den Daten verwendet wird, und <code>soll</code> jeweils der Name des Attributs ist, dessen Funktion das Attribut mit dem Namen <code>ist</code> haben soll.

Mit dem Tag `<pstyle name=beispiel ...>` werden nicht nur das Anfangs- und Ende-Tag `<beispiel>` und `</beispiel>` definiert, sondern auch das leere Tag `<beispiel/>`; letzteres ist gleichbedeutend mit `<beispiel></beispiel>` (ohne Daten zwischen den beiden Tags). Falls jedoch mit dem Attribut `data` z. B. die Zeichenfolge `xy` vorgegeben wurde, ist das Tag `<beispiel/>` gleichbedeutend mit `<beispiel>xy</beispiel>`.

Beispiel für Absatz-Tags für Überschriften

Hauptüberschrift

Zwischenüberschrift

Hier wird die erste Zeile nicht eingerückt, weil der vorangehende Absatz mit einem Tag gekennzeichnet ist, das mit dem Attribut "next-first=left" definiert ist.

Hier wird die erste Zeile wie in der Definition des Tags <pe> angegeben um 15 Punkt eingerückt. Das Attribut "next-first=left" wirkt sich jeweils nur auf den unmittelbar folgenden Absatz aus.

```
<pstyle name=h1 before=24 after=0 size=16  
no-break=title next-first=left hyphenation=off navigation=1/>
```

```
<pstyle name=h2 before=12 after=6 size=14  
no-break=title next-first=left hyphenation=off navigation=2/>
```

```
<pstyle name=pe first=15/>
```

```
<h1>Hauptüberschrift</h1>
```

```
<h2>Zwischenüberschrift</h2>
```

```
<pe>Hier wird die erste Zeile nicht eingerückt, weil der  
vorangehende Absatz mit einem Tag gekennzeichnet ist,  
das mit dem Attribut "next-first=left" definiert ist.</pe>
```

```
<pe>Hier wird die erste Zeile wie in der Definition des  
Tags <pe> angegeben um 15 Punkt eingerückt.  
Das Attribut "next-first=left" wirkt sich jeweils nur auf  
den unmittelbar folgenden Absatz aus.</pe>
```

Tags für Listen

<lstyle/> Definition eines Listen-Tags (list style)

Für dieses Tag ist folgendes Attribut obligat:

name = ... Name des neuen Tags.

Ein mit `<lstyle/>` definiertes Tag kann in gleicher Weise verwendet werden, wie das vordefinierte Tag `` (siehe Seite 32). Es sind automatisch auch alle Attribute möglich, die für das Tag `` vorgesehen sind.

Sollen für das neue Tag bestimmte Attributwerte voreingestellt werden, so können diese im Tag `<lstyle/>` angegeben werden. Für das Attribut `bm` kann jedoch keine Voreinstellung angegeben werden.

Sollen für das vordefinierte Tag `` andere Attributwerte voreingestellt werden, kann es mit dem Tag `<lstyle/>` umdefiniert werden.

Zusätzlich sind im Tag `<lstyle/>` die folgenden Attribute möglich:

<code>pstyle</code>	= *	Angegebene Attribute verwenden. [*]
	= n1'n2'...	Auf Level 1, 2, ... für die Formatierung nicht die im Tag <code><lstyle/></code> angegebenen Attribute verwenden, sondern die Attribute des selbst definierten Absatz-Tags mit dem Namen n, wobei die Attribute <code>first</code> und <code>left</code> ggf. ignoriert werden. Sollen auf einem Level die im Tag <code><lstyle/></code> angegebenen Attribute verwendet werden, kann an Stelle eines Namens ein Stern angegeben werden.
<code>first</code>	= p1'p2'...	Aufzählzeichen auf Level 1 um p1 Punkt einrücken, ab Level 2 jeweils um p Punkt weiter als den Text des nächst kleineren Levels einrücken. [0]
<code>left</code>	= p1'p2'...	Text auf Level 1 um p1 Punkt einrücken, ab Level 2 jeweils um p Punkt weiter einrücken. [Standard-Tabulator]
	= !p1'p2'...	Text auf Level n um p Punkt einrücken.
<code>start</code>	= ...	Zeichenfolge, die unmittelbar nach dem Anfangs-Tag in den Text eingefügt wird.
<code>end</code>	= ...	Zeichenfolge, die unmittelbar vor dem Ende-Tag in den Text eingefügt wird.
<code>data</code>	= ...	Zeichenfolge, die an Stelle des leeren Tags in den Text eingefügt wird.
<code>ignore</code>	= name1'name2'...	Namen der Attribute, die ignoriert werden sollen, wenn sie im neuen Tag vorkommen.
<code>rename</code>	= ist1:soll1'ist2:soll2'...	Namen von Attributen, wobei <code>ist</code> jeweils der Name ist, der in den Daten verwendet wird, und <code>soll</code> jeweils der Name des Attributs ist, dessen Funktion das Attribut mit dem Namen <code>ist</code> haben soll.

<code>label</code>	= <code>z1'z2'...</code>	Angabe, welche Aufzählzeichen für die einzelnen Level der Liste verwendet werden sollen, siehe nachfolgende Übersicht. [<code> . ' -</code>]
<code>label-fgc</code>	= <code>c</code>	Aufzählzeichen in Farbe <code>c</code> . [<code>black</code>]

Die folgende Übersicht zeigt die möglichen Angaben zum Attribut `label`:

.		Malpunkt
-		Minuszeichen
o		Kringel
*		Stern
0	1	Nummer, bei 0 bzw. 1 beginnend
0)	1)	Nummer mit Klammer, bei 0 bzw. 1 beginnend
0.	1.	Nummer mit Punkt, bei 0 bzw. 1 beginnend
.0	.1	Punkt mit Nummer, bei 0 bzw. 1 beginnend (s. u.)
i	I	Römische Zahl in Klein- bzw. Großbuchstaben
a	A	Kleinbuchstabe bzw. Großbuchstabe
a)	A)	Klein- bzw. Großbuchstabe mit Klammer

Soll das Aufzählungszeichen nicht zentriert um die mit dem Attribut `first` vorgegebene Position gedruckt werden, sondern *rechts* oder *links* von dieser Position, so muss unmittelbar *vor* bzw. *nach* der zuvor beschriebenen Angabe zum Attribut `label` ein senkrechter Strich angegeben.

Die Angaben `.0` und `.1` sind nur unmittelbar nach einer der Angaben `0`, `1`, `.0`, `.1`, `a`, `A`, `i` und `I` erlaubt. Sie werden intern mit diesen vorangehenden Angaben kombiniert. So wird z. B. mit

```
<lstyle name=beispiel label=A'.1'.1>
```

auf dem Level drei eine Nummerierung in der Form "A.1.2" erreicht. Bei solchen kombinierten Nummerierungen ist es in der Regel notwendig, mit dem Attribut `left` entsprechend großen Platz für die Nummerierung vorzusehen, da er nicht automatisch angepasst wird.

Wird in einem mit `<lstyle>` definierten Tag ein höherer Level verwendet, als in der Definition Werte mit den Attributen `label`, `first` oder `left` angegeben sind, so wird der letzte angegebene Wert des betreffenden Attributs verwendet. Diese Regelung gilt nicht für das Attribut `pstyle`.

Jedes Listenelement im Text mit dem Level `n`, setzt alle Nummern mit einem Level größer als `n` auf 0 bzw. 1 zurück. Darüber hinaus ist es nicht möglich, in einer Liste die Nummer zurückzusetzen und wieder neu beginnen zu lassen. In MS-Word scheint dies möglich zu sein. Dabei wird aber intern automatisch eine weitere Liste angelegt. Deshalb muss eine weitere Liste mit einem anderen Namen verwendet werden, wenn die Nummerierung an einer Stelle wieder neu beginnen soll, an der sie nicht automatisch neu beginnt. Eine andere Möglichkeit ist, die Nummern mit dem Tag `<num/>` (siehe Seite 66) selber zu verwalten.

Beispiel für nummerierte Listen mit Einrücken

1. Erster Haupteintrag
 2. Zweiter Haupteintrag
 - a) Erster Untereintrag
 - b) Zweiter Untereintrag
 - c) Dritter Untereintrag
 3. Dritter Haupteintrag
 - a) Erster Untereintrag
 - b) Zweiter Untereintrag
-

```
<pstyle name=nuli-1 before=3 size=14/>
<pstyle name=nuli-2 before=3 size=12/>
<lstyle name=nuli label=|1.'|a) left=20 pstyle=nuli-1'nuli-2/>

<nuli>Erster Haupteintrag</nuli>

<nuli>Zweiter Haupteintrag</nuli>
<nuli level=2>Erster Untereintrag</nuli>
<nuli level=2>Zweiter Untereintrag</nuli>
<nuli level=2>Dritter Untereintrag</nuli>

<nuli>Dritter Haupteintrag</nuli>
<nuli level=2>Erster Untereintrag</nuli>
<nuli level=2>Zweiter Untereintrag</nuli>
```

Alternative Version mit kürzerer Formulierung und gleichem Ergebnis:

```
<pstyle name=nuli-1 before=3 size=14/>
<pstyle name=nuli-2 before=3 size=12/>
<lstyle name=nuli label=|1.'|a) left=20 pstyle=nuli-1'nuli-2/>

<alias alias=h1 tag=~nuli level=1/>
<alias alias=h2 tag=~nuli level=2/>

<h1>Erster Haupteintrag</h1>

<h1>Zweiter Haupteintrag</h1>
<h2>Erster Untereintrag</h2>
<h2>Zweiter Untereintrag</h2>
<h2>Dritter Untereintrag</h2>

<h1>Dritter Haupteintrag</h1>
<h2>Erster Untereintrag</h2>
<h2>Zweiter Untereintrag</h2>
```

Die Tags <pstyle> und <lstyle> sind auf Seite 95 bzw. 97, das Tag <alias> ist auf Seite 92 beschrieben.

Beispiel für nummerierte Listen ohne Einrücken

A Erstes Kapitel

A.1 Erste Überschrift

...

A.2 Zweite Überschrift

...

B Zweites Kapitel

B.1 Erste Überschrift

...

B.2 Zweite Überschrift

...

B.3 Dritte Überschrift

...

```
<pstyle name=numü-1 before=18 after=0 size=18 no-break=title/>
<pstyle name=numü-2 before=14 after=6 size=14 no-break=title/>
<lstyle name=numü label=|A|.1 left=!30 pstyle=numü-1'numü-2/>
```

```
<alias alias=h1 tag=~numü level=1/>
<alias alias=h2 tag=~numü level=2/>
```

```
<h1>Erstes Kapitel</h1>
```

```
<h2>Erste Überschrift</h2>
```

```
<p>. . .</p>
```

```
<h2>Zweite Überschrift</h2>
```

```
<p>. . .</p>
```

```
<h1>Zweites Kapitel</h1>
```

```
<h2>Erste Überschrift</h2>
```

```
<p>. . .</p>
```

```
<h2>Zweite Überschrift</h2>
```

```
<p>. . .</p>
```

```
<h2>Dritte Überschrift</h2>
```

```
<p>. . .</p>
```

Die Tags <pstyle> und <lstyle> sind auf Seite 95 bzw. 97, das Tag <alias> ist auf Seite 92 beschrieben.

Tags für Zeichenfolgen

<cstyle/> Definition eines Zeichenfolgen-Tags (character style)

Für dieses Tag ist folgendes Attribut obligat:

name = ... Name des neuen Tags.

Ein mit `<cstyle/>` definiertes Tag kann in gleicher Weise verwendet werden, wie das vordefinierte Tag `<cs>` (siehe Seite 36). Es sind automatisch auch alle Attribute möglich, die für das Tag `<cs>` vorgesehen sind.

Sollen für das neue Tag bestimmte Attributwerte voreingestellt werden, so können diese im Tag `<cstyle/>` angegeben werden.

Sollen für das vordefinierte Tag `<cs>` andere Attributwerte voreingestellt werden, kann es mit dem Tag `<cstyle/>` umdefiniert werden.

Zusätzlich sind im Tag `<cstyle/>` die folgenden Attribute möglich:

start = ... Zeichenfolge, die unmittelbar nach dem Anfangs-Tag in den Text eingefügt wird.

end = ... Zeichenfolge, die unmittelbar vor dem Ende-Tag in den Text eingefügt wird.

data = ... Zeichenfolge, die an Stelle des leeren Tags in den Text eingefügt wird.

ignore = name1'name2'... Namen der Attribute, die ignoriert werden sollen, wenn sie im neuen Tag vorkommen.

rename = ist1:soll1'ist2:soll2'... Namen von Attributen, wobei `ist` jeweils der Name ist, der in den Daten verwendet wird, und `soll` jeweils der Name des Attributs ist, dessen Funktion das Attribut mit dem Namen `ist` haben soll.

Beim Verwenden eines mit `<cstyle>` definierten Tags gilt für das Attribut `style` folgende Regelung: Wird als Attributwert nur ein Stern angegeben oder als erster Teilwert statt eines Namens ein Stern angegeben, so werden die für das Tag voreingestellten Schriftformen übernommen, andernfalls werden sie gelöscht und es werden nur die jeweils angegebenen eingestellt.

Mit dem Tag `<cstyle name=beispiel>` werden nicht nur das Anfangs-Tag `<beispiel>` und das Ende-Tag `</beispiel>` definiert, sondern auch das leere Tag `<beispiel/>`. Letzteres kann nur sinnvoll verwendet werden, wenn in der Definition zum Attribut 'data' eine nicht leere Zeichenfolge angegeben ist; andernfalls ist das Tag `<beispiel/>` wirkungslos.

Beispiel für Zeichenfolgen-Tags

Text mit zwei **gelb markierten** Wörtern, einem *kursiven* Wort, einem unterstrichenen Wort und drei *kursiven und unterstrichenen* Wörter.

```
<cstyle name=marker bgc=yellow/>
```

```
<cstyle name=hvgh style=i char-space=+1/>
```

```
<p>Text mit zwei <marker>gelb markierten</marker> Wörtern,  
einem <hvgh>kursiven</hvgh> Wort,  
einem <hvgh style=u1>unterstrichenen</hvgh> Wort und  
drei <hvgh style=*'u1>kursiven und unterstrichenen</hvgh>  
Wörter.</p>
```

Makros

<macro> Definition eines Makros

Für dieses Tag ist folgendes Attribut obligat:

name = ... Name des Makros.

Weitere Attribute sind nach Bedarf frei wählbar:

Das Makro `<macro>` kann Text, vordefinierte und zuvor selbst definierte Tags enthalten. Sollen an bestimmten Stellen (z. B. für Attributwerte, aber auch für Textteile) erst beim Aufruf des Makros aktuelle Werte eingesetzt werden, so kann dafür ein Stellvertreter in die Definition eingesetzt werden. Ein Stellvertreter ist ein in geschweifte Klammern eingeschlossener Name. Jeder Stellvertreter muss mit dem gleichen Namen als Attribut im Tag `<macro>` angegeben werden; dabei muss als Attributwert der Wert angegeben werden, der für das Attribut voreingestellt sein soll.

Mit dem Tag `<macro>` können gleichzeitig ein leeres Tag sowie ein Anfangs- und Endtag mit dem zum Attribut `name` angegebenen Namen definiert werden. Werden die Daten zwischen `<macro>` und `</macro>` nicht mit dem leeren Tag `<macro/>` unterteilt, so wird nur ein leeres Tag definiert.

Beispiel für Makros für leere Tags

Makrobeispiel

...

Hier ein Siehe-Verweis ohne Makro: (siehe "Makrobeispiel" Seite 103).

Hier einer mit Hilfe eines Makros: (siehe "Makrobeispiel" Seite 103).

```
<macro name=siehe buma="???">
(siehe "<tref bm="{buma}"/>" Seite <pref bm="{buma}"/>) </macro>
```

```
<p bm=beispiel>Makrobeispiel</p>
<p>. . .</p>
<p>Hier ein Siehe-Verweis ohne Makro:
(siehe "<tref bm=beispiel/>" Seite <pref bm=beispiel/>).</p>
<p>Hier einer mit Hilfe eines Makros:
<siehe buma=beispiel/>.</p>
```

Die Tags <tref/> und <pref/> sind auf Seite 62 bzw. 60 beschrieben.

Werden die Daten zwischen `<macro>` und `</macro>` mit dem leeren Tag `<macro/>` unterteilt, so wird mit dem ersten Teil das leere Tag, mit dem zweiten und dritten das Anfangs- und das Ende-Tag mit dem im Tag `<macro>` zum Attribut `name` angegebenen Namen definiert.

Beispiel für Makros für Anfangs- und Ende-Tags

...

Nr.: 123

Autor: Müller, Peter

Titel: Die Dolomiten

...

```

<macro name=eintrag nr=0><macro/>
<p first=0 left=40>Nr.:<tab/>{nr}</p></macro>

<macro name=autor><macro/>
<p first=0 left=40>Autor:<tab/><macro/></p></macro>

<macro name=titel><macro/>
<p first=0 left=40>Titel:<tab/><macro/></p></macro>

<p>. . .</p>
<eintrag nr=123>
  <autor>Müller, Peter</autor>
  <titel>Die Dolomiten</titel>
</eintrag>
<p>. . .</p>

```

Code-Tabellen

<codes/> Definition einer Code-Tabelle

Für dieses Tag ist folgendes Attribut obligat:

`name` = ... Name der Code-Tabelle.

Mit weiteren Attributen kann jeweils ein Code für ein Zeichen definiert werden:

<code>xxxx</code>	= <code>z</code>	Das in TUSTEP mit <code>z</code> codierte Zeichen soll in das Unicode-Zeichen mit dem hexadezimalen Code <code>xxxx</code> umcodiert werden.
	= <code>digits</code>	Die Zeichen, die für die Ziffern 0 bis 9 verwendet werden sollen (z. B. Mediävalziffern), stehen im jeweiligen Font ab dem Code <code>xxxx</code> .
	= <code>lowercase</code>	Die Zeichen, die für die Kleinbuchstaben a bis z verwendet werden sollen (z. B. Kapitälchen), stehen im jeweiligen Font ab dem Code <code>xxxx</code> .
	= <code>uppercase</code>	Die Zeichen, die für die Großbuchstaben A bis Z verwendet werden sollen, stehen im jeweiligen Font ab dem Code <code>xxxx</code> .
	= <code>both-cases</code>	Die Zeichen, die für die Großbuchstaben A bis Z und die Kleinbuchstaben a bis z verwendet werden sollen, stehen im jeweiligen Font ab dem Code <code>xxxx</code> .
	= <code>dagesh</code>	Hebräische Buchstaben mit Dagesch nicht in <code>FBxy</code> , sondern in <code>05xy</code> und <code>xxxx</code> umcodieren, wobei <code>xxxx</code> der Code für den Dagesch ist; nähere Erläuterung siehe unten.
<code>xxxx-yyyy</code>	= <code>z</code>	Das in TUSTEP mit <code>z</code> codierte Zeichen soll in die beiden Unicode-Zeichen mit den hexadezimalen Codes <code>xxxx</code> und <code>yyyy</code> umcodiert werden.

Der Code `xxxx` muss mit einer vierstelligen Hexadezimalzahl angegeben werden. Für `z` kann auch ein Zeichen angegeben werden, das in TUSTEP mit mehr als einem Zeichen codiert wird, z. B. `%/a` für ein `a` mit Akut.

Sollen verschiedene Codierungen in TUSTEP mit dem gleiche Zeichen eines Fonts dargestellt werden, können diese Codierungen gemeinsam in einem Attribut angegeben werden. Die einzelnen Codierungen müssen dabei durch ein Leerzeichen voneinander getrennt werden. Der Apostroph kann hier nicht als Trennzeichen verwendet werden, da er selbst Bestandteil einer Codierung sein kann.

Hebräische Buchstaben werden in TUSTEP mit lateinischen Kleinbuchstaben, hebräische Buchstaben mit Dagesch mit lateinischen Großbuchstaben codiert. Beim Exportieren werden erstere in den Unicode-Bereich `05D0` bis `05EA`, letztere in den Bereich `FB30` bis `FB4A` umcodiert. Da sich in manchen Fonts die Buchstaben aus dem Bereich `FB30` bis `FB40` nicht mit weiteren Punktierungszeichen kombinieren lassen, können auch die Großbuchstaben in den Bereich `05D0` bis `05EA` umcodiert und mit einem Code für den Dagesch versehen werden. Dies wird erreicht, indem der Code für den Dagesch (in der Regel mit `05BC=dagesh`) angegeben wird. Ob es sinnvoll ist, alle Buchstaben mit Dage-

sch in den Bereich 05D0 bis 05EA umzucodieren, hängt vom verwendeten Font ab, da in diesem Fall der Dagesch nicht immer optimal in den Buchstaben positioniert wird. Eventuell ist es sinnvoll nur einzelne Buchstaben entsprechend umzucodieren, z. B. das Alef mit dem Attribut 05D0-05BC=A.

Die Namen der so definierten Code-Tabellen können in Tags für Zeichenfolgen (siehe Seite 36) mit dem Attribut codes angegeben werden. Es ist darauf zu achten, dass jeweils auch ein Font eingestellt ist, für den die angegebene Code-Tabelle zutrifft.

Word enthält keine entsprechende Funktion zum Umcodieren von Zeichen. Das Umcodieren erfolgt beim Exportieren der Daten. Deshalb sind Zeichenfolgen, die erst von Word eingefügt werden (z. B. Seitenzahlen auf Grund des Tags <pref/>), vom Umcodieren nicht betroffen.

„Normale“ Ziffern:	1234567890 (Versalziffern, Tabellenziffern)
Mediävalziffern:	□□□□□□□□□□
„Normale“ Kapitälchen:	BEISPIEL (verkleinerte Großbuchstaben)
Echte Kapitälchen:	B□□□□□□□

```
<codes name=mediäval E020=digits/>
```

```
<codes name=kapit E051=lowercase
E070=%\a E071=%/a E074=ä E077=%;c
E078=%\e E079=%/e E086=ö E08C=ü E091=ß>
```

```
<p font="Linux Libertine G" tabs=|130'|200>
<br/>#.,Normale#.' Ziffern: <tab/>1234567890
<tab/>(Versalziffern, Tabellenziffern)
<br/>Mediävalziffern: <tab/><cs codes=mediäval>1234567890</cs>
<br/>#.,Normale#.' Kapitälchen: <tab/><c>Beispiel</c>
<tab/>(verkleinerte Großbuchstaben)
<br/>Echte Kapitälchen: <tab/><cs codes=kapit>Beispiel</cs></p>
```

Formeln

Eine Formel kann im laufenden Text oder als einziger Inhalt eines Absatz-Tags angegeben werden. Sie muss mit folgendem Tag gekennzeichnet sein:

<mf> Mathematische Formel

Für dieses Tag sind keine Attribute vorgesehen.

Das Tag darf Formeltext oder eine Komponente oder eine Folge von beiden enthalten.

Formeltext besteht aus Variablennamen, Zahlen und mathematischen Zeichen. Er kann auch Klammern enthalten, ihre Größe wird jedoch nicht an ihren Inhalt angepasst. Deshalb ist es in der Regel sinnvoll, Klammern mit der Klammer-Komponente (siehe Seite 110) zu codieren, wenn sie nicht nur Formeltext enthalten.

Komponenten gibt es folgende: Bruch, Hoch/Tiefstellung, Klammern, Wurzel, Summe, Produkt, Integral, Limes, Strich, Akzent, und Funktionsname. Sie sind ab der nächsten Seite beschrieben.

Formeltext kann mit folgendem Tag gekennzeichnet sein:

<mf-t> Formeltext

Für dieses Tag sind folgende Attribute möglich:

<code>font</code>	<code>= ...</code>	Font name verwenden.
	<code>= *</code>	Gleichen Font wie vor der Formel verwenden.
<code>size</code>	<code>= p</code>	Schriftgröße p Punkt.
<code>style</code>	<code>= none</code>	Buchstaben rekte.
	<code>= i</code>	Buchstaben kursiv (italic). [*]
	<code>= b</code>	Alles fett, Buchstaben rekte (bold).
	<code>= b' i</code>	Alles fett, Buchstaben kursiv.
<code>up</code>	<code>= p</code>	Text um p Punkt hochstellen.
<code>dn</code>	<code>= p</code>	Text um p Punkt tiefstellen.

Wenn das Attribut `font` nicht angegeben ist, wird ein zur Umgebung der Formel passender mathematischer Font verwendet.

Wenn das Attribut `font` angegeben ist, ist die Voreinstellung für das Attribut `style` nicht wie angegeben `i`, sondern `none`.

Das Tag `<mf-t>` ist erforderlich, wenn der Formeltext in einer bestimmten Schriftgröße oder einem bestimmten Schriftschnitt dargestellt werden soll. Wenn es nicht angegeben ist, wird es automatisch ergänzt: Die Schriftgröße wird der Umgebung angepasst, die Buchstaben werden kursiv, die Ziffern und Sonderzeichen rekte gedruckt.

Bruch

Ein Bruch kann mit folgendem Tag codiert werden:

<mf-frac> Bruch (fraction)

Für dieses Tag ist folgendes Attribut möglich:

type	= bar	Bruch in der Form $\frac{x}{y}$ [*]
	= linear	Bruch in der Form x/y
	= skewed	Bruch in der Form $\frac{x}{y}$
	= nobar	Bruch in der Form $\frac{x}{y}$

Das Tag <mf-frac> muss folgende zwei Tags enthalten:

<mf-num> Zähler (numerator)

<mf-den> Nenner (denominator)

Andere Daten darf das Tag <mf-frac> nicht enthalten. Die Tags <mf-num> und <mf-den> dürfen Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-num> und <mf-den> ist folgendes Attribut möglich:

size = p Schriftgröße p Punkt.

Beispiel für Brüche

$$x = \frac{a + b}{2}$$

```
<p>
  <mf>
    x=
    <mf-frac>
      <mf-num>a+b</mf-num>
      <mf-den>2</mf-den>
    </mf-frac>
  </mf>
</p>
```

Hoch/Tiefstellung

Hoch/Tiefstellung kann mit folgendem Tag codiert werden:

<mf-s> Hoch/Tiefstellung (superscript, subscript)

Für dieses Tag ist kein Attribut vorgesehen.

Das Tag <mf-s> kann folgende drei Tags enthalten, wobei das erste obligat ist:

<mf-e> Variablenname (element)

<mf-sup> Hochgestellter Teil (superscript)

<mf-sub> Tiefgestellter Teil (subscript)

Andere Daten darf das Tag <mf-s> nicht enthalten. Die Tags <mf-e>, <mf-sup> und <mf-sub> dürfen Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-e>, <mf-sup> und <mf-sub> ist folgendes Attribut möglich:

size = p Schriftgröße p Punkt.

Beispiel für Hoch/Tiefstellungen

$$a^2 + b^2 = c^2 \qquad x = x_1 + x_2 + \dots$$

```
<p><tab pos=30%/>
<mf>
  <mf-s>
    <mf-e>a</mf-e> <mf-sup>2</mf-sup>
  </mf-s>
  +
  <mf-s>
    <mf-e>b</mf-e> <mf-sup>2</mf-sup>
  </mf-s>
  =
  <mf-s>
    <mf-e>c</mf-e> <mf-sup>2</mf-sup>
  </mf-s>
</mf>
<tab pos=60%/>
<mf>
  x =
  <mf-s>
    <mf-e>x</mf-e> <mf-sub>1</mf-sub>
  </mf-s>
  +
  <mf-s>
    <mf-e>x</mf-e> <mf-sub>2</mf-sub>
  </mf-s>
</mf>
  + . . .
</p>
```

Klammern

Werden Klammern z. B. als (und) in die Formel eingefügt, so werden sie in ihrer Größe nicht an ihren Inhalt angepasst. Mit folgendem Tag codierte Klammern werden angepasst:

<mf-d> Klammern (delimiter)

Für dieses Tag sind folgende Attribute möglich:

type	= ()	Runde Klammern. [*]
	= []	Eckige Klammern.
	= { }	Geschweifte Klammern.
	= {	Geschweifte Klammer auf.
	= }	Geschweifte Klammern zu.
	=	Senkrechte Striche.

Das Tag <mf-d> darf Formeltexte und Komponenten enthalten (siehe Seite 107).

Beispiel für Klammern

$$x = \left(\frac{a+b+c}{3}\right)^2 \qquad x = \left|\frac{a+b}{2}\right|$$

```

<p><tab pos=30%/>
<mf>
  x =
  <mf-s>
    <mf-e>
      <mf-d>
        <mf-frac>
          <mf-num>a+b+c</mf-num> <mf-den>3</mf-den>
        </mf-frac>
      </mf-d>
    </mf-e>
    <mf-sup>2</mf-sup>
  </mf-s>
</mf>
<tab pos=60%/>
<mf>
  x =
  <mf-d type=||>
    <mf-frac>
      <mf-num>a+b</mf-num> <mf-den>2</mf-den>
    </mf-frac>
  </mf-d>
</mf></p>

```

Die Tags <mf-s> und <mf-e> sind auf Seite 109 beschreiben,
 <mf-frac>, <mf-num> und <mf-den> sind auf Seite 108 beschrieben.

Wurzel

Eine Wurzel kann mit folgendem Tag codiert werden:

<mf-rad> Wurzel aus ... (radical)

Für dieses Tag ist folgendes Attribut möglich:

size = p Größe des Wurzelzeichens mindestens p Punkt.

Das Tag <mf-rad> kann folgende zwei Tags enthalten, wobei das zweite obligat ist:

<mf-deg> Wurzelexponent (degree)

<mf-e> Wurzel aus ... (element)

Andere Daten darf das Tag <mf-rad> nicht enthalten. Die Tags <mf-deg> und <mf-e> dürfen Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-deg> und <mf-e> ist folgendes Attribut möglich:

size = p Schriftgröße p Punkt.

Vereinfachung: Wenn das Tag <mf-deg> nicht angegeben ist und das Tag <mf-e> nur Formeltext und keine Tags enthält, können die Tags <mf-e> und </mf-e> weggelassen werden; sie werden in diesem Fall automatisch ergänzt.

Beispiel für Wurzeln

$$x = \sqrt[3]{a + b}$$

$$x = \sqrt{a + b}$$

```
<p><tab pos=60%/>
  <mf>
    x =
    <mf-rad>
      <mf-deg>3</mf-deg>
      <mf-e>a + b</mf-e>
    </mf-rad>
  </mf>
<tab pos=60%/>
  <mf>
    x =
    <mf-rad>a + b</mf-rad>
  </mf>
</p>
```

Summe

Eine Summe kann mit folgendem Tag codiert werden:

<mf-S> Summe

Für dieses Tag sind folgende Attribute möglich:

type	= sup-sub	Obere Grenze oberhalb, untere Grenze unterhalb des Summenzeichens drucken. [*]
	= right	Obere und untere Grenze rechts des Summenzeichens drucken.
size	= p	Summenzeichen p Punkt.

Das Tag <mf-S> kann folgende drei Tags enthalten, wobei das zweite obligat ist:

<mf-sup> Obere Grenze (superscript)

<mf-e> Summe über ... (element)

<mf-sub> Untere Grenze (subscript)

Andere Daten darf das Tag <mf-S> nicht enthalten. Die Tags <mf-e>, <mf-sup> und <mf-sub> dürfen Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-e>, <mf-sup> und <mf-sub> ist folgendes Attribut möglich:

size	= p	Schriftgröße p Punkt.
------	-----	-----------------------

Beispiel für Summen

$$\sum_{i=1}^n i = 1 + 2 + \dots + n$$

```

<p>
  <mf>
    <mf-S>
      <mf-sup>n</mf-sup>
      <mf-e>i</mf-e>
      <mf-sub>i=1</mf-sub>
    </mf-S>
    = 1 + 2 + ... + n
  </mf>
</p>

```

Produkt

Ein Produkt kann mit folgendem Tag codiert werden:

<mf-P> Produkt

Für dieses Tag sind folgende Attribute möglich:

type	= sup-sub	Obere Grenze oberhalb, untere Grenze unterhalb des Produktzeichens drucken. [*]
	= right	Obere und untere Grenze rechts des Produktzeichens drucken.
size	= p	Produktzeichen p Punkt.

Das Tag <mf-P> kann folgende drei Tags enthalten, wobei das zweite obligat ist:

<mf-sup> Obere Grenze (superscript)

<mf-e> Produkt von ... (element)

<mf-sub> Untere Grenze (subscript)

Andere Daten darf das Tag <mf-P> nicht enthalten. Die Tags <mf-e>, <mf-sup> und <mf-sub> dürfen Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-e>, <mf-sup> und <mf-sub> ist folgendes Attribut möglich:

size	= p	Schriftgröße p Punkt.
------	-----	-----------------------

Beispiel für Produkte

$$\prod_{i=1}^n i = n! = 1 * 2 * \dots * n$$

```

<p>
  <mf>
    <mf-P>
      <mf-sup>n</mf-sup>
      <mf-e>i</mf-e>
      <mf-sub>i=1</mf-sub>
    </mf-P>
    = n! = 1 * 2 * ... * n
  </mf>
</p>

```

Integral

Ein Integral kann mit folgendem Tag codiert werden:

<mf-I> Integral

Für dieses Tag sind folgende Attribute möglich:

type	= sup-sub	Obere Grenze oberhalb, untere Grenze unterhalb des Integralzeichens drucken.
	= right	Obere und untere Grenze rechts des Integralzeichens drucken. [*]
size	= p	Integralzeichen p Punkt.

Das Tag <mf-I> kann folgende drei Tags enthalten, wobei das zweite obligat ist:

<mf-sup> Obere Grenze (superscript)
<mf-e> Integral über ... (element)
<mf-sub> Untere Grenze (subscript)

Andere Daten darf das Tag <mf-I> nicht enthalten. Die Tags <mf-e>, <mf-sup> und <mf-sub> dürfen Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-e>, <mf-sup> und <mf-sub> ist folgendes Attribut möglich:

size = p Schriftgröße p Punkt.

Beispiel für Integrale

$$A = \int_a^b f(x) dx$$

```
<p>
  <mf>
    A=
    <mf-I>
      <mf-sup>b</mf-sup>
      <mf-e>f(x) dx</mf-e>
      <mf-sub>a</mf-sub>
    </mf-I>
  </mf>
</p>
```

Limes

Limes kann mit einem der beiden folgenden Tag codiert werden, je nachdem, ob die Laufvariable oben oder unten stehen soll:

<mf-lim-upp> Laufvariable oben

<mf-lim-low> Laufvariable unten

Für beide Tags ist folgendes Attribut möglich:

`size` = `p` Schriftgröße für "lim" `p` Punkt.

Das Tag `<mf-lim-upp>` und das Tag `<mf-lim-low>` können folgende zwei Tags enthalten, wobei das zweite obligat ist:

<mf-e> Bezeichnung (element)

<mf-lim> Laufvariable (limes)

Andere Daten dürfen die Tags `<mf-lim-upp>` und `<mf-lim-low>` nicht enthalten. Das Tag `<mf-lim>` darf Formeltext, das Tag `<mf-e>` darf Formeltexte und Komponenten enthalten (siehe Seite 107).

Das Tag `<mf-e>` ist nur erforderlich, wenn statt der Bezeichnung `lim` etwas anderes in die Formel eingesetzt werden soll.

Für die Tags `<mf-e>` und `<mf-lim>` ist folgendes Attribut möglich:

`size` = `p` Schriftgröße `p` Punkt.

Beispiel für Limes

$$\lim_{x \rightarrow \infty} \frac{x}{x + 1} = 1$$

```
<p>
  <mf>
    <mf-lim-low>
      <mf-lim>x#..#(ue)</mf-lim>
    </mf-lim-low>
    <mf-frac>
      <mf-num>x</mf-num>
      <mf-den>x + 1</mf-den>
    </mf-frac>
    = 1
  </mf>
</p>
```

Die Tags `<mf-frac>`, `<mf-num>` und `<mf-den>` sind auf Seite 108 beschrieben.

Funktionsname

Ein Funktionsname kann mit folgendem Tag codiert werden:

<mf-func> Funktion

Für dieses Tag sind keine Attribute vorgesehen.

Das Tag <mf-func> muss die beiden folgenden Tags enthalten:

<mf-name> Funktionsname

<mf-e> Funktion von ... (element)

Andere Daten darf das Tag <mf-func> nicht enthalten. Das Tag <mf-name> darf einen Namen oder eine Hoch/Tiefstellungs-Komponente, das Tag <mf-e> darf Formeltexte und Komponenten enthalten (siehe Seite 107).

Für die Tags <mf-name> und <mf-e> ist folgendes Attribut möglich:

size = p Schriftgröße p Punkt.

Beispiel für Funktionsnamen

$$\sin \alpha = \cos(90^\circ - \alpha)$$

$$\log_b g \cdot \log_g b = 1$$

```

<p><tab pos=20%/>
  <mf>
    <mf-func>
      <mf-name>sin</mf-name>
      <mf-e>#g+a#g-</mf-e>
    </mf-func>
  =
  <mf-func>
    <mf-name>cos</mf-name>
    <mf-e>(90#.*-#g+a#g-)</mf-e>
  </mf-func>
</mf>
<tab pos=70%/>
<mf>
  <mf-func>
    <mf-name><mf-s>log<mf-sub>b</mf-sub></mf-s></mf-name>
    <mf-e>g</mf-e>
  </mf-func>
  ^
  <mf-func>
    <mf-name><mf-s>log<mf-sub>g</mf-sub></mf-s></mf-name>
    <mf-e>b</mf-e>
  </mf-func>
  = 1
</mf>
</p>

```

Strich

Ein Strich über/unter Variablen kann mit folgendem Tag codiert werden:

<mf-bar> Strich

Für dieses Tag sind folgende Attribute möglich:

type = top Strich oben. [*]

= bottom Strich unten.

size = p Schriftgröße p Punkt.

Das Tag <mf-bar> darf Formeltexte und Komponenten enthalten (siehe Seite 107).

Beispiel für Striche

... OBEN ... UNTEN ...

```
<p>
<mf>
  . . .
  <mf-bar>OBEN</mf-bar>
  . . .
  <mf-bar type=bottom>UNTEN</mf-bar>
  . . .
</mf>
</p>
```

Akzent

Ein "Akzent" kann mit folgendem Tag codiert werden:

<mf-acc> Akzent (accent)

Für dieses Tag sind folgende Attribute möglich:

type	= bridge	Brücke. [*]
	= l-arrow	Pfeil nach links.
	= r-arrow	Pfeil nach rechts.
	= lr-arrow	Pfeil nach links und rechts.
	= (Runde Klammer oben.
	=)	Runde Klammer unten.
	= {	Geschefite Klammer oben.
	= }	Geschefite Klammer unten.
size	= p	Schriftgröße p Punkt.

Das Tag <mf-acc> darf Formeltexte und Komponenten enthalten (siehe Seite 107).

Soll "nur" ein Strich gedruckt werden, kann das Tag <mf-bar> (siehe Seite 117) verwendet werden.

Beispiel für Akzente

... BRUECKE ... LINKS ... BEIDE ... RECHTS ... oben ... unten ...

```

<p>
<mf>
  . . .
  <mf-acc>BRUECKE</mf-acc>
  . . .
  <mf-acc type=l-arrow>LINKS</mf-acc>
  . . .
  <mf-acc type=lr-arrow>BEIDE</mf-acc>
  . . .
  <mf-acc type=r-arrow>RECHTS</mf-acc>
  . . .
  <mf-acc type=(>oben</mf-acc>
  . . .
  <mf-acc type=}>unten</mf-acc>
  . . .
</mf>
</p>

```

Beispiel

Beispiel für Formeln

Für eine allgemeine quadratische Gleichung der Form $ax^2 + bx + c = 0$ ergeben sich für x im reellen Zahlenraum zwei Lösungen:

$$x_{1,2} = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

`<p no-break=title>`Für eine allgemeine quadratische Gleichung der Form

```
<mf>
  a
  <mf-s>
    <mf-e>x</mf-e>
    <mf-sup>2</mf-sup>
  </mf-s>
  +bx+c=0
</mf>
```

ergeben sich für `<mf>x</mf>` im reellen Zahlenraum zwei Lösungen:`</p>`

```
<p before=12>
<mf>
  <mf-s>
    <mf-e>x</mf-e>
    <mf-sub>1,2</mf-sub>
  </mf-s>
  =
  <mf-frac>
    <mf-num>
      -b#(plm)
      <mf-rad>
        <mf-e>
          <mf-s>
            <mf-e>b</mf-e>
            <mf-sup>2</mf-sup>
          </mf-s>
          -4ac
        </mf-e>
      </mf-rad>
    </mf-num>
    <mf-den>
      2a
    </mf-den>
  </mf-frac>
</mf>
</p>
```

Vordefinierte Tags

<!-- ... -->	Daten ignorieren	51															
<table border="1"> <tr> <td>---</td> <td>Keine Attribute</td> <td>--</td> </tr> </table>			---	Keine Attribute	--												
---	Keine Attribute	--															
<alias/>	Definition eines Alias-Namens für Tags	92															
<table border="1"> <tr> <td>alias</td> <td>Alias-Name</td> <td>92</td> </tr> <tr> <td>tag</td> <td>Tag-Name, ~Tag-Name</td> <td>92</td> </tr> <tr> <td>...</td> <td>Attribute des angegebenen Tags</td> <td>--</td> </tr> <tr> <td>ignore</td> <td>Attributnamen</td> <td>93</td> </tr> <tr> <td>rename</td> <td>Attributnamenpaare</td> <td>93</td> </tr> </table>			alias	Alias-Name	92	tag	Tag-Name, ~Tag-Name	92	...	Attribute des angegebenen Tags	--	ignore	Attributnamen	93	rename	Attributnamenpaare	93
alias	Alias-Name	92															
tag	Tag-Name, ~Tag-Name	92															
...	Attribute des angegebenen Tags	--															
ignore	Attributnamen	93															
rename	Attributnamenpaare	93															
<alias-defs/>	Tag zur Strukturierung der Definitionen	89															
<table border="1"> <tr> <td>...</td> <td>Beliebige Attribute</td> <td>--</td> </tr> </table>			...	Beliebige Attribute	--												
...	Beliebige Attribute	--															
<am/>	Anmerkungsmarke (annotation mark)	45															
<table border="1"> <tr> <td>name</td> <td>Name</td> <td>45</td> </tr> <tr> <td>id</td> <td>Nummer</td> <td>45</td> </tr> <tr> <td>type</td> <td>start, end</td> <td>45</td> </tr> </table>			name	Name	45	id	Nummer	45	type	start, end	45						
name	Name	45															
id	Nummer	45															
type	start, end	45															
<an>	Anmerkung (annotation)	45															
<table border="1"> <tr> <td>am</td> <td>Name, Nummer</td> <td>45</td> </tr> </table>			am	Name, Nummer	45												
am	Name, Nummer	45															
<an-p/>	Absatz-Tag für Anmerkungstext	46															
<table border="1"> <tr> <td>...</td> <td>Attribute wie <p></td> <td>133</td> </tr> </table>			...	Attribute wie <p>	133												
...	Attribute wie <p>	133															
	fett (bold)	39															
<table border="1"> <tr> <td>---</td> <td>Keine Attribute</td> <td>--</td> </tr> </table>			---	Keine Attribute	--												
---	Keine Attribute	--															
<bl>	<sub>, <sup> aufheben	39															
<table border="1"> <tr> <td>---</td> <td>Keine Attribute</td> <td>--</td> </tr> </table>			---	Keine Attribute	--												
---	Keine Attribute	--															
<bm/>	Lesezeichen (bookmark)	59															
<table border="1"> <tr> <td>name</td> <td>Name</td> <td>59</td> </tr> <tr> <td>id</td> <td>Nummer</td> <td>59</td> </tr> <tr> <td>type</td> <td>start, end</td> <td>59</td> </tr> </table>			name	Name	59	id	Nummer	59	type	start, end	59						
name	Name	59															
id	Nummer	59															
type	start, end	59															
<body>	Text	14															
<table border="1"> <tr> <td>...</td> <td>Beliebige Attribute</td> <td>--</td> </tr> </table>			...	Beliebige Attribute	--												
...	Beliebige Attribute	--															
<bquote/>	Left low single quotation mark (baseline)	15															
<table border="1"> <tr> <td>---</td> <td>Keine Attribute</td> <td>--</td> </tr> </table>			---	Keine Attribute	--												
---	Keine Attribute	--															
 	Umbruch (break)	35															
<table border="1"> <tr> <td>type</td> <td>[line], column, page</td> <td>35</td> </tr> </table>			type	[line], column, page	35												
type	[line], column, page	35															

<code><cstyle/></code>	Definition eines Zeichenfolgen-Tags (character style)	101																																																																								
	<table border="1"> <tr><td>name</td><td>Name</td><td>101</td></tr> <tr><td>start</td><td>Zeichenfolge</td><td>101</td></tr> <tr><td>end</td><td>Zeichenfolge</td><td>101</td></tr> <tr><td>data</td><td>Zeichenfolge</td><td>101</td></tr> <tr><td>ignore</td><td>Attributnamen</td><td>101</td></tr> <tr><td>rename</td><td>Attributnamenpaare</td><td>101</td></tr> <tr><td>font</td><td>Fontname</td><td>36</td></tr> <tr><td>size</td><td>p, ±p</td><td>36</td></tr> <tr><td>kerning</td><td>p</td><td>36</td></tr> <tr><td>scale</td><td>n% [100%]</td><td>36</td></tr> <tr><td>char-space</td><td>±p [0]</td><td>36</td></tr> <tr><td>width</td><td>p</td><td>36</td></tr> <tr><td>style</td><td>[*], i, b, c, C, s, ul, ...</td><td>36</td></tr> <tr><td>bgc</td><td>c</td><td>36</td></tr> <tr><td>fgc</td><td>c</td><td>36</td></tr> <tr><td>ulc</td><td>c [black]</td><td>36</td></tr> <tr><td>border</td><td>[none], box</td><td>36</td></tr> <tr><td>border-color</td><td>c [black]</td><td>36</td></tr> <tr><td>border-width</td><td>p</td><td>36</td></tr> <tr><td>border-space</td><td>p [7]</td><td>36</td></tr> <tr><td>up</td><td>p</td><td>36</td></tr> <tr><td>dn</td><td>p</td><td>36</td></tr> <tr><td>hyphenation</td><td>off</td><td>36</td></tr> <tr><td>codes</td><td>none, Code-Tabellen-Name</td><td>36</td></tr> </table>	name	Name	101	start	Zeichenfolge	101	end	Zeichenfolge	101	data	Zeichenfolge	101	ignore	Attributnamen	101	rename	Attributnamenpaare	101	font	Fontname	36	size	p, ±p	36	kerning	p	36	scale	n% [100%]	36	char-space	±p [0]	36	width	p	36	style	[*], i, b, c, C, s, ul, ...	36	bgc	c	36	fgc	c	36	ulc	c [black]	36	border	[none], box	36	border-color	c [black]	36	border-width	p	36	border-space	p [7]	36	up	p	36	dn	p	36	hyphenation	off	36	codes	none, Code-Tabellen-Name	36	
name	Name	101																																																																								
start	Zeichenfolge	101																																																																								
end	Zeichenfolge	101																																																																								
data	Zeichenfolge	101																																																																								
ignore	Attributnamen	101																																																																								
rename	Attributnamenpaare	101																																																																								
font	Fontname	36																																																																								
size	p, ±p	36																																																																								
kerning	p	36																																																																								
scale	n% [100%]	36																																																																								
char-space	±p [0]	36																																																																								
width	p	36																																																																								
style	[*], i, b, c, C, s, ul, ...	36																																																																								
bgc	c	36																																																																								
fgc	c	36																																																																								
ulc	c [black]	36																																																																								
border	[none], box	36																																																																								
border-color	c [black]	36																																																																								
border-width	p	36																																																																								
border-space	p [7]	36																																																																								
up	p	36																																																																								
dn	p	36																																																																								
hyphenation	off	36																																																																								
codes	none, Code-Tabellen-Name	36																																																																								
<code><cstyle-defs/></code>	Tag zur Strukturierung der Definitionen	89																																																																								
	<table border="1"> <tr><td>...</td><td>Beliebige Attribute</td><td>--</td></tr> </table>	...	Beliebige Attribute	--																																																																						
...	Beliebige Attribute	--																																																																								
<code><date/></code>	Datum einfügen	56																																																																								
	<table border="1"> <tr><td>---</td><td>Keine Attribute</td><td>--</td></tr> </table>	---	Keine Attribute	--																																																																						
---	Keine Attribute	--																																																																								
<code><df></code>	dünn gestrichelt (dashed, fine)	40																																																																								
	<table border="1"> <tr><td>ulc</td><td>c [black]</td><td>40</td></tr> </table>	ulc	c [black]	40																																																																						
ulc	c [black]	40																																																																								
<code><Df></code>	dünn gestrichelt-punktiert (dash-dotted, fine)	40																																																																								
	<table border="1"> <tr><td>ulc</td><td>c [black]</td><td>40</td></tr> </table>	ulc	c [black]	40																																																																						
ulc	c [black]	40																																																																								
<code><document></code>	Dokument	14																																																																								
	<table border="1"> <tr><td>...</td><td>Beliebige Attribute</td><td>--</td></tr> </table>	...	Beliebige Attribute	--																																																																						
...	Beliebige Attribute	--																																																																								
<code><dstyle/></code>	Voreinstellungen (document style / default style)	19																																																																								
	<table border="1"> <tr><td>template</td><td>Dateipfad</td><td>19</td></tr> <tr><td>mirror</td><td>[off], on</td><td>19</td></tr> <tr><td>distance</td><td>[both], maximum</td><td>19</td></tr> </table>	template	Dateipfad	19	mirror	[off], on	19	distance	[both], maximum	19																																																																
template	Dateipfad	19																																																																								
mirror	[off], on	19																																																																								
distance	[both], maximum	19																																																																								

line-break	expand	19
tabs	p [18]	19
footnote	1, [2], i, ii, I, II, a, b, A, B	19
endnote	1, 2, i, [ii], I, II, a, b, A, B	19
left	p [65]	20
right	p [65]	20
top	p [60]	20
bottom	p [60]	20
header	p [36]	20
footer	p [36]	20
pagination	[auto], none	20
border	[none], text, page	20
border-color	c [black]	20
border-width	p	20
border-space	p [5]	20
align	left, center, right, [both], justify	20
font	<i>Fontname</i> [Times]	20
size	p [12]	20
kerning	p	20
hyphenation	[on], off	20
<dt>	dick gestrichelt (dashed, thick)	40
ulc	c [black]	40
<Dt>	dick gestrichelt-punktiert (dash-dotted, thick)	40
ulc	c [black]	40
<dummy>	Tag ignorieren	52
...	<i>Beliebige Attribute</i>	--
<E>	Relief (embossed)	39
---	<i>Keine Attribute</i>	--
<en>	Endnote	42
char	<i>Zeichen</i>	42
auto-tabs	before, [after], n [1]	42
bm	<i>Name</i>	42
<en-c>	Zeichenfolgen-Tag für Endnotennummern	42
...	<i>Attribute wie <cs></i>	121
<en-p/>	Absatz-Tag für Endnotentext	42
...	<i>Attribute wie <p></i>	133
<emdash/>	Strich, etwa ein ganzes Geviert lang	15
---	<i>Keine Attribute</i>	--

<emspace/>	Spatium, etwa ein ganzes Geviert breit	15
---	<i>Keine Attribute</i>	--
<endash/>	Strich, etwa ein halbes Geviert lang	15
---	<i>Keine Attribute</i>	--
<enspace/>	Spatium, etwa ein halbes Geviert breit	15
---	<i>Keine Attribute</i>	--
<export-defs>	Defintionen für *EXPORT	89
...	<i>Beliebige Attribute</i>	--
<field/>	<i>Feld einfügen</i>	88
inst	<i>Feldanweisung</i>	88
rslt	<i>Feldinhalt</i>	88
<fill/>	Text an den rechten Rand schieben, Zwischenraum punktieren	55
---	<i>Keine Attribute</i>	--
<fn>	Fußnote	42
char	<i>Zeichen</i>	42
auto-tabs	before, [after], n [1]	42
bm	<i>Name</i>	42
<fn-c>	Zeichenfolgen-Tag für Fußnotennummern	42
...	<i>Attribute wie <cs></i>	121
<fn-p/>	Absatz-Tag für Fußnotentext	42
...	<i>Attribute wie <p></i>	133
	Schriftart	37
name	<i>Fontname</i>	37
size	p, ±p	37
kerning	p	37
scale	n% [100%]	37
char-space	±p [0]	37
width	p	37
style	[*], i, b, c, C, s, ul, ...	37
up	p	37
dn	p	37
codes	none, <i>Code-Tabellen-Name</i>	37
<fpref/>	Verweis auf den ersten Absatzes (first paragraph reference)	63
pstyle	<i>Name</i>	63
<ftr>	Fußtext (footer)	57
type	first, odd, even, [both]	57

<ftr-p/>	Absatz-Tag für Fußtext	57
...	Attribute wie <p>	133
<h>	verborgen (hidden)	39
---	Keine Attribute	--
<hdr>	Kopftext (header)	57
type	first, odd, even, [both]	57
<hdr-p/>	Absatz-Tag für Kopftext	57
...	Attribute wie <p>	133
<head>	Definitionen	14
...	Beliebige Attribute	--
<hl>	Hyperlink	64
bm	Name, Nummer	64
to	file:///..., http://..., https://...	64
cstyle	Name	64
style	[*], i, b, c, C, s, ul, ...	64
bgc	c	64
fgc	c	64
ulc	c [black]	64
border	[none], box	64
border-color	c [black]	64
border-width	p	64
border-space	p [7]	64
<hr/>	Trennlinie (horizontal rule)	34
left	p [0]	34
right	p [0]	34
no-break	after, [none]	34
mirror	[off], on	34
before	p	34
after	p	34
line-color	c [black]	34
line-width	p	34
<hyphen/>	Strich, etwa ein viertel Geviert lang	15
---	Keine Attribute	--
<i>	kursiv (italic)	39
---	Keine Attribute	--
<I>	Gravur (imprinted)	39
---	Keine Attribute	--

<ignore>	Daten ignorieren	51																																																																																				
	<table border="1"> <tr> <td>...</td> <td><i>Beliebige Attribute</i></td> <td>--</td> </tr> </table>	...	<i>Beliebige Attribute</i>	--																																																																																		
...	<i>Beliebige Attribute</i>	--																																																																																				
<index/>	Register erstellen	81																																																																																				
	<table border="1"> <tr> <td>name</td> <td><i>Name</i></td> <td>81</td> </tr> <tr> <td>add</td> <td>[letter], space, nothing</td> <td>81</td> </tr> <tr> <td>align</td> <td>[none], right</td> <td>81</td> </tr> <tr> <td>cols</td> <td>n</td> <td>81</td> </tr> </table>	name	<i>Name</i>	81	add	[letter], space, nothing	81	align	[none], right	81	cols	n	81																																																																									
name	<i>Name</i>	81																																																																																				
add	[letter], space, nothing	81																																																																																				
align	[none], right	81																																																																																				
cols	n	81																																																																																				
<index-n/>	Absatz-Tags für Registereinträge	83																																																																																				
	<table border="1"> <tr> <td>...</td> <td><i>Attribute wie <p></i></td> <td>133</td> </tr> </table>	...	<i>Attribute wie <p></i>	133																																																																																		
...	<i>Attribute wie <p></i>	133																																																																																				
<lf>	lang dünn gestrichelt (long dashed, fine)	40																																																																																				
	<table border="1"> <tr> <td>ulc</td> <td>c [black]</td> <td>40</td> </tr> </table>	ulc	c [black]	40																																																																																		
ulc	c [black]	40																																																																																				
	Listeneintrag (list item)	32																																																																																				
	<table border="1"> <tr> <td>level</td> <td>n [1]</td> <td>32</td> </tr> <tr> <td>indent</td> <td>p [0]</td> <td>32</td> </tr> <tr> <td>page</td> <td>[same], next,</td> <td>25</td> </tr> <tr> <td>no-break</td> <td>within, after, title, [none]</td> <td>25</td> </tr> <tr> <td>m-pos</td> <td>p [0]</td> <td>25</td> </tr> <tr> <td>right</td> <td>p [0]</td> <td>25</td> </tr> <tr> <td>mirror</td> <td>[off], on</td> <td>25</td> </tr> <tr> <td>before</td> <td>p [0]</td> <td>25</td> </tr> <tr> <td>after</td> <td>p [0]</td> <td>25</td> </tr> <tr> <td>distance</td> <td>[apply], suppress</td> <td>25</td> </tr> <tr> <td>align</td> <td>left, center, right, both, justify, distribute</td> <td>26</td> </tr> <tr> <td>font</td> <td><i>Fontname</i></td> <td>26</td> </tr> <tr> <td>size</td> <td>p</td> <td>26</td> </tr> <tr> <td>kerning</td> <td>p</td> <td>26</td> </tr> <tr> <td>line-feed</td> <td>p, p!</td> <td>26</td> </tr> <tr> <td>scale</td> <td>n% [100%]</td> <td>26</td> </tr> <tr> <td>char-space</td> <td>±p [0]</td> <td>26</td> </tr> <tr> <td>style</td> <td>[*], i, b, c, C, s, ul, ...</td> <td>26</td> </tr> <tr> <td>color</td> <td>c</td> <td>26</td> </tr> <tr> <td>bgc</td> <td>c</td> <td>26</td> </tr> <tr> <td>fgc</td> <td>c</td> <td>26</td> </tr> <tr> <td>ulc</td> <td>c [black]</td> <td>26</td> </tr> <tr> <td>line-number</td> <td>[none]</td> <td>26</td> </tr> <tr> <td>border</td> <td>[none], box, top, bottom</td> <td>26</td> </tr> <tr> <td>border-color</td> <td>c [black]</td> <td>26</td> </tr> <tr> <td>border-width</td> <td>p</td> <td>26</td> </tr> <tr> <td>border-space</td> <td>p [7]</td> <td>27</td> </tr> <tr> <td>bar</td> <td>c</td> <td>27</td> </tr> </table>	level	n [1]	32	indent	p [0]	32	page	[same], next,	25	no-break	within, after, title, [none]	25	m-pos	p [0]	25	right	p [0]	25	mirror	[off], on	25	before	p [0]	25	after	p [0]	25	distance	[apply], suppress	25	align	left, center, right, both, justify, distribute	26	font	<i>Fontname</i>	26	size	p	26	kerning	p	26	line-feed	p, p!	26	scale	n% [100%]	26	char-space	±p [0]	26	style	[*], i, b, c, C, s, ul, ...	26	color	c	26	bgc	c	26	fgc	c	26	ulc	c [black]	26	line-number	[none]	26	border	[none], box, top, bottom	26	border-color	c [black]	26	border-width	p	26	border-space	p [7]	27	bar	c	27	
level	n [1]	32																																																																																				
indent	p [0]	32																																																																																				
page	[same], next,	25																																																																																				
no-break	within, after, title, [none]	25																																																																																				
m-pos	p [0]	25																																																																																				
right	p [0]	25																																																																																				
mirror	[off], on	25																																																																																				
before	p [0]	25																																																																																				
after	p [0]	25																																																																																				
distance	[apply], suppress	25																																																																																				
align	left, center, right, both, justify, distribute	26																																																																																				
font	<i>Fontname</i>	26																																																																																				
size	p	26																																																																																				
kerning	p	26																																																																																				
line-feed	p, p!	26																																																																																				
scale	n% [100%]	26																																																																																				
char-space	±p [0]	26																																																																																				
style	[*], i, b, c, C, s, ul, ...	26																																																																																				
color	c	26																																																																																				
bgc	c	26																																																																																				
fgc	c	26																																																																																				
ulc	c [black]	26																																																																																				
line-number	[none]	26																																																																																				
border	[none], box, top, bottom	26																																																																																				
border-color	c [black]	26																																																																																				
border-width	p	26																																																																																				
border-space	p [7]	27																																																																																				
bar	c	27																																																																																				

dblbar	c	27
hyphenation	[on], off	27
codes	none, <i>Code-Tabellen-Name</i>	27
navigation	[none], n	27
bm	<i>Name</i>	27
tabs	[none], <i>Tabulatordefinitionen</i>	27
<ln-c>	Zeichenfolgen-Tag für Zeilennummern	23
...	<i>Attribute wie <cs></i>	121
<lref/>	Verweis auf den letzten Absatzes (last paragraph reference)	63
pstyle	<i>Name</i>	63
<lquote/>	Left single quotation mark	15
---	<i>Keine Attribute</i>	--
<lref/>	Verweis auf einen Listeneintrag (list item reference)	61
bm	<i>Name, Nummer</i>	61
<lstyle/>	Definition eines Listen-Tags (list style)	97
name	<i>Name</i>	97
pstyle	[*], <i>Namen</i>	97
first	p1'p2,... [0]	97
left	p1'p2,..., !p1'p2,...	97
start	<i>Zeichenfolge</i>	97
end	<i>Zeichenfolge</i>	97
data	<i>Zeichenfolge</i>	97
ignore	<i>Attributnamen</i>	97
rename	<i>Attributnamenpaare</i>	97
label	<i>Aufzählzeichen</i>	98
label-fgc	c [black]	98
page	[same], next,	25
no-break	within, after, title, [none]	25
m-pos	p [0]	25
right	p [0]	25
mirror	[off], on	25
before	p [0]	25
after	p [0]	25
distance	[apply], suppress	25
align	left, center, right, both, justify, distribute	26
font	<i>Fontname</i>	26
size	p	26
kerning	p	26
line-feed	p, p!	26

scale	n% [100%]	26
char-space	±p [0]	26
style	[*], i, b, c, C, s, ul, ...	26
color	c	26
bgc	c	26
fgc	c	26
ulc	c [black]	26
line-number	[none]	26
border	[none], box, top, bottom	26
border-color	c [black]	26
border-width	p	26
border-space	p [7]	27
bar	c	27
dblbar	c	27
hyphenation	[on], off	27
codes	none, <i>Code-Tabellen-Name</i>	27
navigation	[none], n	27
next-pstyle	<i>Name</i>	27
bm	<i>Name</i>	27
tabs	[none], <i>Tabulatordefinitionen</i>	27
<lstyle-defs/>	Tag zur Strukturierung der Definitionen	89
...	<i>Beliebige Attribute</i>	--
<lt>	lang dick gestrichelt (long dashed, thick)	40
ulc	c [black]	40
<macro>	Definition eines Makros	103
name	<i>Name</i>	103
...	<i>Beliebige Attribute</i>	--
<macro/>	Unterteilung einer Makrodefinition	103
---	<i>Keine Attribute</i>	--
<macro-defs/>	Tag zur Strukturierung der Definitionen	89
...	<i>Beliebige Attribute</i>	--
<mf>	Mathematische Formel	107
---	<i>Keine Attribute</i>	--
<mf-acc>	Akzent (accent)	118
type	[bridge], l-arrow, r-arrow, lr-arrow, (,), {, }	118
size	p	118
<mf-bar>	Strich	117
type	[top], bottom	117

	size	p	117
<mf-d>	Klammern (delimiter)		110
	type	() , [] , {} , { , } ,	110
<mf-deg>	Wurzelexponent (degree)		111
	size	p	111
<mf-den>	Nenner (denominator)		108
	size	p	108
<mf-e>	Daten (element)		109 – 116
	size	p	109
<mf-frac>	Bruch (fraction)		108
	type	bar , linear , skewed , nobar	108
<mf-func>	Funktion		116
	---	<i>Keine Attribute</i>	116
<mf-I>	Integral		114
	type	[sup-sub] , right	114
	size	p	114
<mf-lim>	Laufvariable (limes)		115
	size	p	115
<mf-lim-low>	Laufvariable unten		115
	size	p	115
<mf-lim-upp>	Laufvariable oben		115
	size	p	115
<mf-name>	Funktionsname		116
	size	p	116
<mf-num>	Zähler (numerator)		116
	size	p	108
<mf-P>	Produkt		113
	type	[sup-sub] , right	113
	size	p	113
<mf-rad>	Wurzel aus ... (radical)		111
	size	p	111
<mf-s>	Hoch/Tiefstellung (superscript, subscript)		109
	---	<i>Keine Attribute</i>	--

<mf-S>	Summe		112
	type	[sup-sub], right	112
	size	p	112
<mf-sub>	Tiefgestellter Teil (subscript)		109 – 114
	size	p	109
<mf-sup>	Hochgestellter Teil (superscript)		109 – 114
	size	p	109
<mf-t>	Formeltext		107
	font	Fontname, *	107
	size	p	107
	style	[none], i, b, i'b	107
	up	p	107
	dn	p	107
<mn>	Marginalie (marginal note)		47
	width	adjust, p	47
	height	adjust, p	47
	v-pos	±p [-3]	47
	h-pos	±p [-53], left, right, left-side, right-side, inside, outside	47
	textflow	[normal], bottom-top, top-bottom	47
	border	[none], box, ellipse, pointer, l-arrow, r-arrow	47
	border-color	c [black]	48
	border-width	p	48
	border-space	p	48
	color	c	48
	opacity	n% [0%]	48
	option	ignore	48
<mn-p/>	Absatz-Tag für Marginalientext		48
	...	Attribute wie <p>	133
<move/>	Schreibposition verschieben		88
	v-pos	p, ±p	88
	h-pos	p, ±p	88
<name/>	Dokumentname einfügen		56
	---	Keine Attribute	--
<nb>	 aufheben		39
	---	Keine Attribute	--

<nc>	<c> aufheben		39
	---	<i>Keine Attribute</i>	--
<nC>	<C> aufheben		39
	---	<i>Keine Attribute</i>	--
<ndf>	<df> aufheben		40
	---	<i>Keine Attribute</i>	--
<nDf>	<Df> aufheben		40
	---	<i>Keine Attribute</i>	--
<ndt>	<dt> aufheben		40
	---	<i>Keine Attribute</i>	--
<nDt>	<Dt> aufheben		40
	---	<i>Keine Attribute</i>	--
<nE>	<E> aufheben		39
	---	<i>Keine Attribute</i>	--
<nh>	<h> aufheben		39
	---	<i>Keine Attribute</i>	--
<ni>	<i> aufheben		39
	---	<i>Keine Attribute</i>	--
<nI>	<l> aufheben		39
	---	<i>Keine Attribute</i>	--
<nlf>	<lf> aufheben		40
	---	<i>Keine Attribute</i>	--
<nlt>	<lt> aufheben		40
	---	<i>Keine Attribute</i>	--
<nO>	<O> aufheben		39
	---	<i>Keine Attribute</i>	--
<nof>	<of> aufheben		40
	---	<i>Keine Attribute</i>	--
<nOf>	<Of> aufheben		40
	---	<i>Keine Attribute</i>	--
<nospace/>	Optionaler Zeilenwechsel		17
	---	<i>Keine Attribute</i>	--

<not>	<ot> aufheben	40
	---	Keine Attribute
<nOt>	<Ot> aufheben	40
	---	Keine Attribute
<nref/>	Verweis auf eine Fußnote/Endnote (note reference)	60
	bm	Name, Nummer
<ns>	<s> aufheben	39
	---	Keine Attribute
<nS>	<S> aufheben	39
	---	Keine Attribute
<ns1>	<s1> aufheben	40
	---	Keine Attribute
<ns2>	<s2> aufheben	40
	---	Keine Attribute
<ntc/>	Nächste Tabellenzelle (next table cell)	70
	---	Keine Attribute
<nu>	<u1>, <u2>, <ut>, <uw>, <w1>, <w2>, <wt> aufheben	40
	---	Keine Attribute
<nu1>	<u1> aufheben	40
	---	Keine Attribute
<nu2>	<u2> aufheben	40
	---	Keine Attribute
<num/>	Laufende Nummer	66
	name	Name
	value	n, [next], same
	type	[decimal], roman, ROMAN, alphabetic, ALPHABETIC, hidden
	bm	Name
<nut>	<ut> aufheben	40
	---	Keine Attribute
<nuw>	<uw> aufheben	40
	---	Keine Attribute
<nw1>	<w1> aufheben	40
	---	Keine Attribute

<nw2>	<w2> aufheben	40
	---	Keine Attribute
		--
<nwt>	<wt> aufheben	40
	---	Keine Attribute
		--
<O>	Umriss (outlined)	39
	---	Keine Attribute
		--
<of>	dünn unterpunktirt (dotted, fine)	40
	ulc	c [black]
		40
<Of>	dünn punktiert-gestrichelt (dot-dash-dotted, fine)	40
	ulc	c [black]
		40
<ot>	dick unterpunktirt (dotted, thick)	40
	ulc	c [black]
		40
<Ot>	dick punktiert-gestrichelt (dot-dash-dotted, thick)	40
	ulc	c [black]
		40
<p>	Absatz (paragraph)	25
	page	[same], next,
		25
	no-break	within, after, title, [none]
		25
	m-pos	p [0]
		25
	first	p [0]
		25
	left	p [0]
		25
	right	p [0]
		25
	direction	[ltr], rtl
		25
	mirror	[off], on
		25
	before	p [0]
		25
	after	p [0]
		25
	distance	[apply], suppress
		25
	align	left, center, right, both, justify, distribute
		26
	font	Fontname
		26
	size	p
		26
	kerning	p
		26
	line-feed	p, p!
		26
	scale	n% [100%]
		26
	char-space	±p [0]
		26
	style	[*], i, b, c, C, s, ul, ...
		26
	color	c
		26
	bgc	c
		26
	fgc	c
		26
	ulc	c [black]
		26

line-number	[none]	26
border	[none], box, top, bottom	26
border-color	c [black]	26
border-width	p	26
border-space	p [7]	27
bar	c	27
dblbar	c	27
hyphenation	[on], off	27
codes	none, <i>Code-Tabellen-Name</i>	27
navigation	[none], n	27
next-pstyle	<i>Name</i>	27
bm	<i>Name</i>	27
i-char	<i>Buchstabe oder Zeichenfolge</i>	27
i-num	<i>Name</i>	27
i-font	<i>Fontname</i>	27
i-size	p	27
i-space	p	27
i-bgc	c	27
i-fgc	c [black]	27
i-type	[margin], page	27
tabs	[none], <i>Tabulatordefinitionen</i>	27

<page/> **Seitennummer einfügen** **56**

---	<i>Keine Attribute</i>	--
-----	------------------------	----

<picture/> **Bild einfügen** **84**

path	<i>Dateipfad</i>	84
file	<i>Dateiname</i>	84
scale	n% [100%]	84
width-scale	n% [100%]	84
height-scale	n% [100%]	84
width	p	84
height	p	84
v-pos	±p [0]	84
h-pos	±p [0], [left], right	84
border	[none], box	84
border-color	c [black]	84
border-width	p	84
border-space	p [0]	84
text	[around], above, below	84

<pos> **Positionieren** **38**

up	p	38
dn	p	38

	size	p	38
	hyphenation	off	38
<pref/>	Verweis auf eine Seite (page reference)		60
	bm	<i>Name, Nummer</i>	60
<pstyle/>	Definition eines Absatz-Tags (paragraph style)		95
	name	<i>Name</i>	95
	next-first	[first], left	95
	start	<i>Zeichenfolge</i>	95
	end	<i>Zeichenfolge</i>	95
	data	<i>Zeichenfolge</i>	95
	ignore	<i>Attributnamen</i>	95
	rename	<i>Attributnamenpaare</i>	95
	page	[same], next,	25
	no-break	within, after, title, [none]	25
	m-pos	p [0]	25
	first	p [0]	25
	left	p [0]	25
	right	p [0]	25
	direction	[ltr], rtl	25
	mirror	[off], on	25
	before	p [0]	25
	after	p [0]	25
	distance	[apply], suppress	25
	align	left, center, right, both, justify, distribute	26
	font	<i>Fontname</i>	26
	size	p	26
	kerning	p	26
	line-feed	p, p!	26
	scale	n% [100%]	26
	char-space	±p [0]	26
	style	[*], i, b, c, C, s, ul, ...	26
	color	c	26
	bgc	c	26
	fgc	c	26
	ulc	c [black]	26
	line-number	[none]	26
	border	[none], box, top, bottom	26
	border-color	c [black]	26
	border-width	p	26
	border-space	p [7]	27
	bar	c	27

dblbar	c	27
hyphenation	[on], off	27
codes	none, <i>Code-Tabellen-Name</i>	27
navigation	[none], n	27
bm	<i>Name</i>	27
i-char	<i>Buchstabe oder Zeichenfolge</i>	27
i-num	<i>Name</i>	27
i-font	<i>Fontname</i>	27
i-size	p	27
i-space	p	27
i-bgc	c	27
i-fgc	c [black]	27
i-type	[margin], page	27
tabs	[none], <i>Tabulatordefinitionen</i>	27
<code><pstyle-defs/></code>	Tag zur Strukturierung der Definitionen	89
...	<i>Beliebige Attribute</i>	--
<code><right/></code>	Text an den rechten Rand schieben, Zwischenraum leer lassen	55
---	<i>Keine Attribute</i>	--
<code><rquote/></code>	Right single quotation mark	15
---	<i>Keine Attribute</i>	--
<code><s></code>	gesperrt (stretched)	39
---	<i>Keine Attribute</i>	--
<code><S></code>	schattiert (shadowed)	39
---	<i>Keine Attribute</i>	--
<code><s1></code>	einfach durchgestrichen (striked)	40
ulc	c [black]	40
<code><s2></code>	doppelt durchgestrichen (striked)	40
ulc	c [black]	40
<code><section></code>	Absatzgruppe	22
page	next, [same], even, odd	22
page-number	[decimal], roman, ROMAN, n	22
line-number	[none], n, n/section, n/page	22
cols	n [1], p1'p2'...	22
col-space	p [18]	22
col-bar	yes, [no]	22
layout	[portrait], landscape	22
left	p	22
right	p	22

top	p	23
bottom	p	23
header	p	23
footer	p	23
border	[*], none	23
border-color	c [black]	23
border-width	p	23
border-space	p [5]	23
align	[top], center, distribute, bottom	23
bm	<i>Name</i>	23

<sstyle/>

Definition eines Absatzgruppen-Tags (section style)

94

name	<i>Name</i>	94
ignore	<i>Attributnamen</i>	94
rename	<i>Attributnamenpaare</i>	94
page	next, [same], even, odd	22
page-number	[decimal], roman, ROMAN, n	22
line-number	[none], n, n/section, n/page	22
cols	n [1], p1'p2'...	22
col-space	p [18]	22
col-bar	yes, [no]	22
layout	[portrait], landscape	22
left	p	22
right	p	22
top	p	23
bottom	p	23
header	p	23
footer	p	23
border	[*], none	23
border-color	c [black]	23
border-width	p	23
border-space	p [5]	23
align	[top], center, distribute, bottom	23

<sstyle-defs/> Tag zur Strukturierung der Definitionen

89

...	<i>Beliebige Attribute</i>	--
-----	----------------------------	----

<sub/>

tiefgestellt, kleinere Schrift

39

---	<i>Keine Attribute</i>	--
-----	------------------------	----

<sup/>

hochgestellt, kleinere Schrift

39

---	<i>Keine Attribute</i>	--
-----	------------------------	----

<tab/>

Tabulator

53

pos	p, p , p, p,	53
-----	--------------	----

	fill	., ^., -, _	53
<table>	Tabelle		68
	color	c, c1'c2	68
	indent	p	68
	align	left, center, right	68
	grid	none, p, c [black], p:c	68
	col-space	p [8]	68
	row-space	p [4]	69
	header	n [0]	69
	no-break	row	69
	pstyle	Name [tb-p], Namen	69
	cols	Spaltendefinitionen	69
<tb-p/>	Absatz-Tag für Tabellentext		71
	...	Attribute wie <p>	133
<tc>	Tabellenzelle (table cell)		70
	color	c	70
<tf>	Tabellenformel		72
	data	f(x), x+y, x-y, x*y, x/y, x*n%	72
<time/>	Uhrzeit einfügen		56
	---	Keine Attribute	--
<toc>	Verzeichnis erstellen (table of contents)		75
	pstyle	t1:n1't2:n2, ...	75
	name	Name	75
	bm	Name, Nummer	75
<toc-n/>	Absatz-Tags für Verzeichniseinträge		79
	...	Attribute wie <p>	133
<toc-p/>	Absatz-Tag für Verzeichniseinträge		79
	...	Attribute wie <p>	133
<toce>	Eintrag für Verzeichnis (table of contents, entry)		77
	level	n	77
<toct>	Eintrag für Verzeichnis (table of contents, text + entry)		77
	level	n	77
<tr>	Tabellenzeile (table row)		69
	color	c	69
	join	n-m, [n, n]	69
	no-break	within, after	69
	pstyle	Name, Namen	70

	cols	Spaltendefinitionen	70
<trref/>		Verweis auf einen Textteil (text reference)	62
	bm	Name, Nummer	62
<u1>		einfach unterstrichen (underlined)	40
	ulc	c [black]	40
<u2>		doppelt unterstrichen (underlined)	40
	ulc	c [black]	40
<ut>		dick unterstrichen (underlined, thick)	40
	ulc	c [black]	40
<uw>		wortweise unterstrichen (underlined, word)	40
	ulc	c [black]	40
<values/>		Definition eines Namens für einen Attributwert	91
	...	Beliebige Attribute	--
<value-defs/>		Tag zur Strukturierung der Definitionen	89
	...	Beliebige Attribute	--
<w1>		einfach gewellt unterstrichen (waved)	40
	ulc	c [black]	40
<w2>		doppelt gewellt unterstrichen (waved)	40
	ulc	c [black]	40
<wt>		dick gewellt unterstrichen (waved, thick)	40
	---	Keine Attribute	--
<xe>		Eintrag für Register (index entry)	80
	name	Name	80
	ref	none, i, b	80
	bm	Name, Nummer	80
<xr/>		Referenz für Registereintrag (index reference)	80
	---	Keine Attribute	--
<xs/>		Untereintrag für Register (index subentry)	80
	---	Keine Attribute	--
<xt>		Eintrag für Register (index entry + text)	80
	name	Name	80
	ref	none, i, b	80
	bm	Name, Nummer	80

Stichwörter

- *EXPORT 11
- *IMPORT 6
- *TAGUM 9
- [*] 20
- [**] 21
- <!--...--> 51, 89
- Absätze 25, 95
- Absatzgruppen 22, 94
- Abschnitt 25
- Abstand zwischen
 - Absätzen 19, 25
- Alias-Namen 92
- Anführungszeichen 15
- Anmerkungen 45
- Apostroph 15
- Arabisch 41
- Attributwerte 13, 91
- Aufzählungen 32
- Bastelecke 88
- Bilder 84
- Blocksatz 20, 26
- Bookmark *siehe*
 - Lesezeichen
- Code-Tabelle 27, 36, 37, 105
- Dateiname *siehe*
 - Dokumentname
- Daten ignorieren 51
- Datenstruktur 14, 19, 89, 92
- Datum 56
- Definitionen 89
- Dokumentname 56
- durchgestrichen 40
- Endnoten 19, 42
- Farben 18
- fett 39
- Flattersatz 20, 26
- Folgezeilen 25
- Formatvorlagen 90
- Formeln 107
- Fußnoten 19, 42
- Fußtext 57
- gesperrt 39
- Griechisch 41
- Hebräisch 41
- Hochformat 22
- hochstellen 36, 37, 38, 39, 107
- Hyperlinks 64
- Initiale 27
- Kapitälchen 39
- Kerning 20, 26, 36, 37
- Kommentar *siehe*
 - Anmerkungen
- Kopftext 57
- kursiv 39
- Laufende Nummern 66, 92
- Lesezeichen 59
- Listen 25, 32, 97
- Makros 103
- Marginalien 47
- mehrspaltig 22, 73
- Navigation 27, 59
- neue Seite 22, 25
- neue Zeile 35
- Nummerierte Listen 32
- Parallelsatz 73
- PDF-Datei 11, 59, 62
- Phonetisch 41
- Punkt (Maßeinheit) 13
- Querformat 22
- Rand 20, 22, 25
- Register 80
- Russisch 41
- Satzspiegelbreite 20
- Satzspiegelhöhe 20
- schattiert 39
- Schattierung 26
- Schriftformen 26, 36, 37, 39, 40, 64
- Schriftgröße 26, 36, 37, 38
- Schriftumschaltung 41
- Seitennummer 22, 56
- Seitenwechsel 22, 25
- senkrechter Strich 22, 27, 110
- Silbentrennung 17
- Sonderzeichen 15
- Spalten 22
- Spaltenwechsel 35
- Standard-Tabulatoren 19
- Tabellen 68
- Tabellenformel 72
- Tabulatordefinition 27
- Tabulatoren 53
- Tags ignorieren 32, 52
- Textmarker 26
- tiefstellen 36, 37, 38, 39, 107
- Trennlinien 34
- Uhrzeit 56
- Unicode-Zeichen 15
- Unterlegung 40
- unterpunktirt 40
- Unterstreichung 26, 36, 37, 40, 64
- Versalien 39
- Verweise 60, 92
- Verzeichnisse 74
- Voreinstellungen 19, 20, 89
- waagrechte Linie 34, 70
- XML-Datei 5, 62
- Zeichenabstand 26, 36, 37
- Zeichenbreite 26, 36, 37
- Zeichenfolgen 36
- Zeichenvorrat 15
- Zeilenabstand 26
- Zeilennummer 22
- Zeilenwechsel 35
- Zentrieren 55
- Zwangsumbruch 35