

KOMA-Script Datei `scrjura.dtx` *

Markus Kohm[†] Dr. Alexander Willand[‡]

2014/01/28

Das `scrjura`-Paket ist für die Kautelarjurisprudenz gedacht. Es soll für Anwälte und Notare eine flexible Hilfe bei der Abfassung von Verträgen, Satzungen und juristischen Kommentaren bieten. Es ist in Zusammenarbeit mit Dr. Alexander Willand entstanden, bzw. noch immer in Entstehung begriffen.

Inhaltsverzeichnis

1	Rechtlicher Hinweis	2
2	Das Paket »scrjura«	2
2.1	Überblick	2
2.2	Die Optionen beim Laden des Pakets	3
2.3	Optionen betreffend Kolummentitel	5
2.4	Der Paragraph	5
2.5	Verweise	7
2.6	Absatz- und Satznummerierung	9
2.7	kile anpassen, <code>scrjura</code> unter <code>lyx</code>	9
2.8	Sonstiges	10
3	Bekannte Probleme	11
4	Implementierung	11

*Diese Datei ist Version v0.7a von `scrjura.dtx`.

[†]Paketautor, [mailto:komascript\(at\)gmx.info](mailto:komascript(at)gmx.info)

[‡]Autor der Anleitung, [mailto:alexander.willand\(klammeraffe\)t-online.de](mailto:alexander.willand(klammeraffe)t-online.de)

1 Rechtlicher Hinweis

Dieser Teil von KOMA-Script darf ohne die restlichen Teile von KOMA-Script verteilt werden. Dazu dürfen die Dateien »scrlogo.dtx« und »scrjura.dtx« unter den Bedingungen der L^AT_EX Project Public License, wahlweise Version 1.3c oder (nach Ihrer Entscheidung) jede spätere Version, verteilt und verändert werden. Die aktuelle Version dieser Lizenz ist unter <http://www.latex-project.org/lppl.txt> zu finden und Version 1.3c oder neuer ist Teil von L^AT_EX Version 2005/12/01 oder neuer. Es ist nicht gestattet die Datei »scrjura.dtx« ohne die Datei »scrlogo.dtx« zu verteilen. Eine Verteilung ist nur zusammen gestattet.

Zu KOMA-Script existiert eine ausführliche Anleitung in Deutsch und Englisch. Die Anleitung zum Paket scrjura existiert derzeit jedoch nur in dieser deutschen Kurzform.

Das komplette KOMA-Script-Paket ist unter CTAN:/macros/latex/contrib/koma-script/ zu finden, wobei »CTAN:« ein abkürzendes Synonym für das Verzeichnis »tex-archive« jedes CTAN-Servers oder -Spiegels, beispielsweise <ftp://ftp.dante.de/tex-archive/> darstellt. Siehe <http://www.ctan.org> für eine Liste aller CTAN-Server und -Spiegel.

2 Das Paket »scrjura«

2.1 Überblick

Das Paket scrjura ist in erster Linie für den Satz von Verträgen gedacht. Über eine Reihe von Optionen kann die Wirkung des Pakets verändert werden. Innerhalb von Verträgen steht eine Reihe von Befehlen zur Verfügung.

Wer juristische Dokumente verfasst, steht oft unter hohem Zeitdruck. Überraschende Fehlermeldungen von L^AT_EX 2_ε kann man da nicht brauchen. Erfreulicherweise setzt scrjura lediglich das scrkbase-Paket voraus.

scrjura bietet zwei Modi:

1. den „Vertragsmodus“: das gesamte Dokument wird als Vertrag gesetzt;
2. den „Umgebungsmodus“: Nur der Text innerhalb der contract-Umgebung wird als Vertrag gesetzt.

Der *Vertragsmodus* wird durch die Option `contract` beim Laden des Pakets dauerhaft eingeschaltet. Also: Mit `\usepackage[contract]{scrjura}` wird Text nach `\begin{document}` als Vertrag gesetzt. Der *Umgebungsmodus* wird dagegen wie jede andere Umgebung mit `\begin{contract}` begonnen und ist entsprechend mit `\end{contract}` zu beenden. Beide Modi können nicht im gleichen Dokument verwendet werden.

Ein Inhaltsverzeichnis wird wie üblich mit `\tableofcontents` erzeugt und enthält – neben etwaigen anderen Überschriften – auch alle Paragraphentitel, sofern `scrjura` mit der Option `juratotoc` geladen wurde. Automatische Kolumnentitel sind ebenfalls möglich.

Jeder Paragraph hat einen Titel, z. B.: `\Paragraph{title=Besitz}`. Die Nummerierung der Paragraphen erfolgt automatisch, kann aber auch manuell vorgenommen werden.

Nahezu alle vom Anwender anzugebenden Daten folgen dem *key=value*-Schema, wie bereits an dem eben genannten Titel eines Paragraphen zu sehen war.

Enthält ein Paragraph mehr als einen Absatz, werden alle Absätze automatisch nummeriert. Die Nummerierung kann fallweise abgeschaltet werden. Es ist darüber hinaus möglich, die Sätze eines Absatzes nummerieren zu lassen.

Paragraphen, Absätze und Sätze können referenziert werden. Bei Absätzen besteht die Wahl, wie die Ausgabe erfolgt; es ist sowohl möglich, beispielsweise „§ 16 Absatz 3 Satz 1“ als auch „§ 16 III 1“ als Ausgabe zu erhalten.

`scrjura` unterstützt das `hyperref`-Paket, ist damit aber hier noch nicht ausführlich getestet worden. Wie immer, wird `hyperref` zuletzt geladen. Andernfalls gibt es faszinierende Fehlermeldungen. Wird `hyperref` wieder entfernt, muss die `*.aux`-Datei gelöscht werden.

2.2 Die Optionen beim Laden des Pakets

<code>contract</code>	Die Option <code>contract</code> sorgt dafür, dass das gesamte Dokument als Vertrag interpretiert wird. Innerhalb des Dokuments ist dann keine weitere Vertragsumgebung zu verwenden. Dasselbe Verhalten kann erreicht werden, indem unmittelbar nach <code>\begin{document}</code> eine <code>contract</code> -Umgebung begonnen und unmittelbar vor <code>\end{document}</code> wieder beendet wird.
<code>juratitlepagebreak</code>	Normalerweise sind Seitenumbrüche innerhalb der Überschriften von Paragraphen nicht erlaubt. Einige Juristen verwenden aber so lange Überschriften, dass selbst Seitenumbrüche darin sinnvoll sein können. Mit der

Option `juratitlepagebreak` wird dies ermöglicht. Diese Option kann jederzeit mit Hilfe von `\KOMAOptions{juratitlepagebreak}` ein- oder mit `\KOMAOptions{juratitlepagebreak=false}` wieder ausgeschaltet werden.

`juratotoc`
`juratoclevel` Die Option `juratotoc` sorgt dafür, dass die Paragraphen in das Inhaltsverzeichnis eingetragen werden. Dabei wird in der Voreinstellung die Gliederungstiefe 2 für diese Einträge verwendet. Über den Zähler `juratoclevel` kann auch eine andere Gliederungstiefe vorgegeben werden. Das Paket greift hier auf die Routinen von L^AT_EX zur Erstellung eines Inhaltsverzeichnisses zurück. Bei dreistelligen Paragraphennummern kann im Inhaltsverzeichnis die Nummer in den Überschriftstext hineinragen.

Die einfachste Abhilfe gegen diesen Fehler besteht darin, die Box mit der Gliederungsnummer zu verbreitern, beispielsweise mit

```
\renewcommand{\numberline}[1]{\makebox[6em][l]{#1}}
```

die Breite der Box den jeweiligen Gegebenheiten anzupassen.

Eine andere Abhilfe bei der Verwendung einer KOMA-Script-Klasse besteht darin, die Option `tocleft` zu verwenden. Zuletzt kann man selbstverständlich auf die verschiedenen Pakete zurückgreifen, mit denen sich das Inhaltsverzeichnis anpassen lässt. Den Gewinn an Möglichkeiten erkauft man sich allerdings mit einem erhöhten Risiko, dass Fehler auftreten.

Ein für den Autor der Anleitung entscheidender Vorteil von `scrjura` liegt in dem umfassenden System, auf andere Klauseln des Vertrages verweisen zu können. Beim Laden des Paktes kann per Option entschieden werden, wie die Verweise formatiert werden sollen: lang (Bsp: »§ 314 Absatz 2 Satz 2«), kurz (»§ 314 Abs. 2 S. 2«) und numerisch (»§ 314 II 2«). Die Optionen hierzu lauten:

- `ref=numeric`
- `ref=long`
- `ref=short`

Wird nichts angegeben, ist `ref=long` voreingestellt.

Wer hier spezifische Wünsche hat, kann die Zitierweise von Absätzen und Sätzen bereits bei den Optionen hiervon abweichend festlegen. Es sind beliebige Kombinationen von `parlong`, `parshort`, `parnumeric`, `nopar` mit `sentenceslong`, `sentenceshort`, `sentencesnumeric`, `nosentence` möglich. Beispiel: `\usepackage[ref=parlong,ref=sentenceshort]{scrjura}`. Wir haben allerdings diese Kombinationen nicht alle getestet.

2.3 Optionen betreffend Kolummentitel

Bei langen Verträgen können automatische Kolummentitel hilfreich sein. Allerdings hängt die Ausgabe hier davon ab, wie die diese erzeugt werden. `scrjura` unterstützt lebende Kolummentitel. Die Option `paragraphmark` kann auf folgende Werte gesetzt werden:

- `false`, `off`, `no`
- `right`
- `both`

Sollte der Vertrag in Abschnitte unterteilt werden, also etwa mit einer Zeile `\parnumberfalse\section{ABSCHNITT}\parnumbertrue`, würde auf den ungeraden Seiten die Abschnittsüberschriften und auf den geraden Seiten der Titel des aktuellen Paragraphen angegeben werden.

Im übrigen können die zur Gestaltung von Kopfzeilen üblichen Pakete (z.B. `scrpage2`) verwendet werden.

2.4 Der Paragraph

`scrjura` führt zwei Gliederungsebenen ein: Paragraph und Subparagraph. Selbstverständlich funktioniert deren im folgenden beschriebene Befehle im *Umgebungsmodus* nur innerhalb der Vertragsumgebung.

`\Paragraph` Für die Paragraphenüberschrift stehen verschiedene optionale Argumente zur Verfügung, die ebenfalls in einer durch Kommata zu trennenden `key=value`-Liste angegeben werden:

Eintrag	Erklärung	Eintrag unterdrücken
<code>title = <Titel></code>	Titel des Paragraphen	<code>dummy</code>
<code>head = ...</code>	Kolummentitel des Paragraphen	<code>nohead</code>
<code>entry = ...</code>	Eintrag im Inhaltsverzeichnis	<code>noentry</code>
<code>number= ...</code>	die Paragraphennummer	-

Beispiele: `\Paragraph{title=Gattungsschuld, number=244}` gibt „§ 244 Gattungsschuld“ aus. `\Paragraph{title=Gattungsschuld}` als erster Paragraph gibt „§ 1 Gattungsschuld“ aus, während `\Paragraph{}` an gleicher Stelle zu „§ 1“ führt, also einer Paragraphenzählung ohne Titel.

Will man aber beginnend von beispielsweise § 244 an eine Reihe von Paragraphen setzen, empfiehlt es sich nicht, die Option `number=` zu verwenden, denn die Option lässt den Zähler `Paragraph` unberührt. Will man nicht alle Paragraphen per Hand nummerieren, kann mit `\setcounter{Paragraph}{243}` der Startpunkt für eine automatische Zählung gesetzt werden. Bei `\SubParagraph` ist der Zähler natürlich mit einer arabischen Zahl anzugeben, auch wenn die Ausgabe dann in Kleinbuchstaben erfolgt.

`\SubParagraph` Sofern ein Paragraph eingeschoben werden soll, kann dieses Makro verwendet werden, es verhält sich im übrigen wie `\Paragraph`. Die Nummerierung fügt der zuletzt verwendeten Paragraphennummer einen Kleinbuchstaben hinzu.

Die Nummerierung für `\SubParagraph` erfolgt in `scrjura.sty` über folgende Definition:

```
\renewcommand*{\thecontractSubParagraph}{%
{\theParagraph\alph{contractSubParagraph}}}
```

Wer zwischen Paragraphenzahl und dem Kleinbuchstaben einen Abstand wünscht, sowie nach dem Kleinbuchstaben eine runde Klammer, kann in die Präambel also einfügen:

```
\renewcommand*{\thecontractSubParagraph}{%
{\theParagraph~\alph{contractSubParagraph}}).
```

`preskip` Abstände vor und nach der Paragraphen-Überschrift können derzeit über
`postskip` `\setkeys{contract}{preskip=... , postskip=...}` gesetzt werden.

Es ist möglich, einen Vertrag in Abschnitte zu unterteilen unter Verwendung z.B. von `\section{}`. Allerdings muss entweder vorher die `contract`-Umgebung beendet und danach fortgesetzt werden (die Paragraphenzählung wird dadurch nicht zurückgesetzt), oder man setzt davor ein `\parnumberfalse` und danach ein `\parnumbertrue`. Vergisst man dies, wird der Abschnitt zum ersten als Paragraphenabsatz und dann als Abschnitt nummeriert.

`\ellipsispar` Manchmal wird insbesondere für Kommentare gewünscht, dass man Absätze in Paragraphen auch auslassen kann, diese Absätze aber dennoch mitgezählt und auch bei der automatischen Absatznummerierung mitberücksichtigt werden. Die ausgelassenen Absätze sollen dabei in der Regel durch Pünktchen markiert werden. Genau dies erledigt `\ellipsispar`. Als optionales Argument kann dabei die Anzahl der Absätze angegeben werden, die ausgelassen wurde. Voreingestellt ist ein Absatz. Für das Auslassungszeichen wird `\parellipsis` verwendet. Dies ist mit `\textellipsis` vordefiniert, falls eine solche Anweisung existiert. Anderenfalls wird `\dots` verwendet.

2.5 Verweise

Verlässlich funktionierende Verweise sind eine große Erleichterung beim Abfassen von Verträgen. Wenn in einem ersten Entwurf eines Gesellschaftsvertrags die Abfindungsregelungen in §§ 28 ff. enthalten sind, diese aber nach einigen Umbauten sich irgendwann in §§ 19 ff. wiederfinden, dann müssen alle Verweise auf die unterschiedlichen Abfindungsregelungen für kündigende Gesellschafter, für solche, die in Rente gehen und für 'rausgeworfene diese Änderungen nachvollzogen haben.

In der Grundeinstellung führt der Verweis `\ref{testA}` auf ein `\label{testA}`, das in einem Absatz platziert wurde, zur Ausgabe eines Vollverweises einschließlich der Satzangabe.

Nun will man häufig nicht den Paragraph, Absatz *und* Satz, sondern nur Paragraph und Absatznummer referenzieren. Die folgende Tabelle enthält eine systematische Darstellung aller Verweismöglichkeiten. Ein »L« kennzeichnet die Langform, ein »S« die Kurzform und ein »N« die numerische Darstellung des Verweises.

Zunächst das Beispiel, auf das sich die Tabelle bezieht:

```
\begin{contract}
\Paragraph{title=Beispiel}
Es irrt der Mensch.\label{testA}
\end{contract}
```

Dies ergibt einen sehr kurzen Paragraphen:

§ 1 Beispiel

Es irrt der Mensch.

Folgende Verweise auf diesen § 1 Absatz 1 Satz 1 sind möglich:

Code	Ergebnis
<code>\ref{testA}</code>	§ 1 paragraph 1 sentence 1
<code>\refL{testA}</code>	§ 1 paragraph 1 sentence 1
<code>\refS{testA}</code>	§ 1 par. 1 sent. 1
<code>\refN{testA}</code>	§ 1 I 1.
<code>\refParagraph{testA}</code>	§ 1
<code>\refParagraphN{testA}</code>	1
<code>\refPar{testA}</code>	paragraph 1
<code>\refParL{testA}</code>	paragraph 1
<code>\refParS{testA}</code>	par. 1
<code>\refParN{testA}</code>	I
<code>\refParN[arabic]{testA}</code>	1
<code>\refSentence{testA}</code>	sentence 1
<code>\refSentenceL{testA}</code>	sentence 1
<code>\refSentenceS{testA}</code>	sent. 1
<code>\refSentenceN{testA}</code>	1.

Dabei hängt die Ausgabe von `\ref`, `\refPar` und `\refSentence` von den Einstellungen der Option `ref` ab. Hier zum Vergleich mit der Einstellung `\KOMAoptions{ref=nopar}`:

Code	Ergebnis
<code>\ref{testA}</code>	§ 1 sentence 1
<code>\refPar{testA}</code>	
<code>\refSentence{testA}</code>	sentence 1

Das gleiche mit `\KOMAoptions{ref=nosentence}`:

Code	Ergebnis
<code>\ref{testA}</code>	§ 1 paragraph 1
<code>\refPar{testA}</code>	paragraph 1
<code>\refSentence{testA}</code>	

Das gleiche mit `\KOMAoption{ref}{nopar,nosentence}`:

Code	Ergebnis
<code>\ref{testA}</code>	§ 1
<code>\refPar{testA}</code>	
<code>\refSentence{testA}</code>	

Zu jeder existiert noch eine Stern-Version, also z.B. `\ref*{testA}`. Wenn das Paket `hyperref` geladen ist, erzeugt die besternte Version einen Verweis, der *kein* Hyperlink ist. Wenn das Paket jedoch nicht geladen wurde, erzeugt die besternte Version – Fehler.

2.6 Absatz- und Satznummerierung

`\pnumbertrue` Absätze in Paragraphen werden automatisch nummeriert. Mit `\pnumberfalse`
`\pnumberfalse` kann die Nummerierung ab- und mit `\pnumbertrue` wieder eingeschaltet werden.

`\Sentence` Auch alle einzelnen Sätze eines Absatzes können durch eine hochgestellte Ziffer nummeriert werden. Allerdings kann dies nicht automatisch geschehen. Am einfachsten gestaltet sich die Satznummerierung unter Zuhilfenahme des Befehls `\useshortands` aus dem ohnehin bei deutschen Texten notwendigen `babel`-Paket:

```
\usepackage[...]{babel}
\useshortands{'}
\defineshortand{'S}{\Sentence\ignorespaces}
\defineshortand{'.'}{. \Sentence\ignorespaces}
```

Die Sätze lassen sich mit einem vorangestellten `'S` nummerieren. Alternativ kann der Satzendeapostroph gesetzt werden: `'.`, damit wird dann der nächste Satz nummeriert.

2.7 kile anpassen, scrjura unter lyx

Bei langen Verträgen ist es sehr hilfreich, die Gliederung und die Paragraphentitel auf dem Bildschirm einblenden zu können. Wer `kile` als Editor verwendet, der kann sich – Open Source sei Dank! – selbst helfen. Bei Version 2.0.0 von `kile` genügt es, in der Datei `kiledocumentinfo.cpp` an geeigneter Stelle zwei Zeilen (jeweils ohne Zeilenumbruch) einzufügen:

```
m_dictStructLevel["\\Paragraph"]=
KileStructData(5, KileStruct::Sect, "Paragraph");

m_dictStructLevel["\\SubParagraph"]=
KileStructData(6, KileStruct::Sect, "SubParagraph");
```

Zahlen über 7 sind eine schlechte Idee. Nach dem Speichern der Datei folgt der übliche Dreisprung: `configure`, `make`, `make install` (oder `checkinstall`

...). kile zeigt danach Paragraphen und »Unterparagraphen« als Dokumentenstruktur an. Vielleicht findet ein Anwender ja heraus, wie dies mit `TeXnicCenter` zu bewerkstelligen ist.

Bis jemand für die Verwendung von `scrjura` eine `lyx`-Klasse schreibt, wozu wir ausdrücklich ermutigen wollen, kann man sich bei der Nutzung von `lyx` damit behelfen, im sog. »`LATEX`-Vorspann« beispielsweise die Definition des Unterunterabschnitts anzupassen:

```
\renewcommand{\subsubsection}[1]{\Paragraph{title= #1}}
```

Damit lassen sich Paragraphenüberschriften in `lyx` als Unterunterabschnittsüberschriften setzen. Da die Absätze von Verträgen ohnehin von `scrjura` automatisch nummeriert werden, zeigen sich die Nachteile dieses Vorgehens erst, wenn man einen Unterunterabschnitt benötigt.

2.8 Sonstiges

<code>\parname</code>	Sprachabhängiger Name von Absatz und Satz.
<code>\sentencename</code>	Ausgabe der automatischen Zähler für Paragraph und eingeschobener Paragraph bei Verträgen.
<code>\thecontractParagraph</code>	
<code>\thecontractSubParagraph</code>	
<code>\AutoPar</code>	Automatische und manuelle Absatznummerierung.
<code>\ManualPar</code>	Absatznummern ein- und ausschalten.
<code>\Par</code>	
<code>\ifparnumber</code>	Zähler für die Absatznummern und dessen Ausgabe und Formatierung und
<code>\parnumbertrue</code>	Formatierung der Referenz. ACHTUNG: Wenn <code>\thepar</code> nicht nur eine arabische Zahl ist, werden Absätze bei automatischer Nummerierung unabhängig davon nummeriert, ob es mehr als einen Absatz im Paragraphen gibt.
<code>\parnumberfalse</code>	
<code>par</code>	
<code>\thepar</code>	
<code>\parformat</code>	So etwas wie <code>\ref</code> nur für den Paragraphen.
<code>\parciteformat</code>	
<code>\refParagraph</code>	So etwas wie <code>\ref</code> nur für den Absatz.
<code>\refPar</code>	
<code>\refSentence</code>	So etwas wie <code>\ref</code> nur für den Satz.
<code>sentence</code>	
<code>\thesentence</code>	Zähler für die Satznummer und dessen Ausgabe, sowie Formatierung der Referenz. ACHTUNG: <code>\thesentence</code> muss bei unverändertem <code>\sentenceciteformat</code> eine arabische Zahl sein!
<code>\sentenceciteformat</code>	
<code>\newmaxpar</code>	Werden intern verwendet und müssen deshalb allenfalls für Paketautoren dokumentiert werden.
<code>\getmaxpar</code>	

3 Bekannte Probleme

Die Verweistechnik wurde noch nicht ausführlich getestet; Fehler sind möglich.

4 Implementierung

```
1 \@ifpackageloaded{hyperref}{%
2   \PackageError{scrjura}{Package hyperref already loaded}{%
3     If you want to use package scrjura with package hyperref, you have to
4     use\MessageBreak
5     package scrjura before package hyperref.\MessageBreak
6     To solve the problem, you just should move the loading of package
7     hyperref\MessageBreak
8     behind the loading of package scrjura.}%
9 }
```

Es wird das Paket scrkbase benötigt, weil alles weitere darauf aufbaut.

```
10 \RequirePackage{scrkbase}[2013/03/26]
```

Außerdem wird tocbasic benötigt, weil einige wenige Befehle davon verwendet werden.

```
11 \RequirePackage{tobasic}
```

`contract` Mit der Option `contract` wird das gesamte Dokument zu einem Vertrag. Man darf dann keine der von diesem Paket bereit gestellten Umgebungen mehr innerhalb des Dokuments verwenden! Realisiert wird dies, indem ganz am Ende von `\begin{document}` noch `\contract` ausgeführt wird.

```
12 \DeclareOption{contract}{%
13   \g@addto@macro\document\contract%
14 }
```

`juratotoc` Mit dieser Option wird der Zähler

`juratoclevel` auf einen gewünschten Wert oder 2 oder 10000 gesetzt.

`\if@juratotoc` Mit dem Zähler wiederum wird erreicht, dass Paragraphen in das Inhaltsverzeichnis aufgenommen werden.
`juratoclevel`

`\toclevel@cpar`

```
15 \newcounter{juratoclevel}\setcounter{juratoclevel}{\@M}
16 \KOMA@key{juratotoc}[true]{%
17 \begingroup
18 \KOMA@set@ifkey{juratoclevel}{@tempswa}{#1}%
19 \ifx\FamilyKeyState\FamilyKeyStateProcessed
20 \if@tempswa
21 \setcounter{juratoclevel}{2}%
22 \else
23 \setcounter{juratoclevel}{\@M}%
24 \fi
25 \else
26 \setcounter{juratoclevel}{#1}%
27 \FamilyKeyStateProcessed
28 \fi
29 \endgroup
30 }
31 \providecommand*\toclevel@cpar{\arabic{juratoclevel}}
```

`juratocnumberwidth` Der Einzug und die Breite der Nummer für Inhaltsverzeichniseinträge kann über diese beiden Optionen bzw. Makros bestimmt werden.

`juratocindent`
`\cpar@numberwidth`
`\cpar@indent`

```
32 \newcommand*\cpar@numberwidth{2em}
33 \newcommand*\cpar@indent{1.5em}
34 \KOMA@key{juratocnumberwidth}{%
35 \begingroup\setlength{\@tempdima}{#1}\endgroup
36 \renewcommand*\cpar@numberwidth{#1}%
37 \FamilyKeyStateProcessed
38 }
39 \KOMA@key{juratocindent}{%
40 \begingroup\setlength{\@tempdima}{#1}\endgroup
41 \renewcommand*\cpar@indent{#1}%
42 \FamilyKeyStateProcessed
43 }
```

`juratitlepagebreak` Mit dieser Option wird der Schalter `\if@juratitlepagebreak` auf `\iftrue` gesetzt.

`\if@juratitlepagebreak` Mit diesem Schalter wiederum wird erreicht, dass in den Überschriften von Paragraphen ein Seitenumbruch erlaubt wird.

```
44 \KOMA@ifkey{juratitlepagebreak}{@juratitlepagebreak}
```

`parnumber` Die (automatische) Absatznummerierung lässt sich nun auch per Option ein- und ausschalten.

```
45 \newif\ifparnumber
```

```

46 \KOMA@key{parnumber}[true]{%
47 \ifstr{#1}{auto}{\AutoPar\FamilyKeyStateProcessed}{%
48 \ifstr{#1}{manual}{\ManualPar\FamilyKeyStateProcessed}{%
49 \KOMA@set@ifkey{parnumber}{parnumber}{#1}%
50 }%
51 }%
52 }

```

`paragraphmark` Mit diesen Optionen wird `\Paragraphmark` so umdefiniert, dass es en-
`markright` tweder ein `\markright` oder ein `\markboth` ausführt. Im Gegensatz zu
`markboth` `\chaptermark` etc. erwartet `\Paragraphmark` neben dem Titel auch noch
`\Paragraphmark` die zu verwendende Nummer (das könnte bei Bedarf noch geändert wer-
den!)

```

53 \newcommand*{\Paragraphmark}[1]{%
54 \KOMA@key{paragraphmark}{%
55 \begingroup
56 \KOMA@set@ncmdkey{paragraphmark}{@tempa}{%
57 {false}{0},{off}{0},{no}{0},%
58 {forceright}{1},%
59 {forceboth}{2},%
60 {right}{3},%
61 {both}{4}%
62 }{#1}%
63 \ifx\FamilyKeyState\FamilyKeyStateProcessed
64 \ifcase\number\@tempa
65 \endgroup
66 \let\Paragraphmark@gobble
67 \or
68 \endgroup
69 \renewcommand*{\Paragraphmark}[1]{%
70 \markright{\csname MakeMarkcase\endcsname{##1}}}%
71 \or
72 \endgroup
73 \renewcommand*{\Paragraphmark}[1]{%
74 \markboth{\csname MakeMarkcase\endcsname{##1}}%
75 {\csname MakeMarkcase\endcsname{##1}}}%
76 \or
77 \endgroup
78 \renewcommand*{\Paragraphmark}[1]{%
79 \ifx
80 \@mkboth\@gobbletwo
81 \else
82 \markright{\csname MakeMarkcase\endcsname{##1}}%
83 \fi}%
84 \or
85 \endgroup
86 \renewcommand*{\Paragraphmark}[1]{%
87 \@mkboth{\csname MakeMarkcase\endcsname{##1}}%

```

```

88         {\csname MakeMarkcase\endcsname{##1}}}%
89     \else
90         \endgroup
91     \fi
92 \else
93     \endgroup
94 \fi
95 }
96 \KOMA@DeclareDeprecatedOption[scrjura]{markright}{paragraphmark=forceright}
97 \KOMA@DeclareDeprecatedOption[scrjura]{markboth}{paragraphmark=forceboth}

```

ref Die Form, in der Absätze und Sätze referenziert werden. Es gibt eine lange,
parcite**name** eine abgekürzte und eine numerische Form.
sentence**name**

\parcite@**format** In der Voreinstellung wird das lange Format verwendet. Es gilt: 0 = lang,
\sentencecite@**format** 1 = kurz, 2 = numerisch, -1 = gar nicht

```

98 \newcommand*{\parcite@format}{0}
99 \newcommand*{\sentencecite@format}{0}

```

Per Option kann die Voreinstellung verändert werden.

```

100 \KOMA@key{ref}{%
101     \begingroup
102         \KOMA@set@ncmdkey{ref}{@tempa}{%
103             {parlong}{1},{longpar}{1},{ParL}{1},%
104             {parshort}{2},{shortpar}{2},{ParS}{2},%
105             {parnumeric}{3},{numericpar}{3},{ParN}{3},%
106             {paroff}{4},{nopar}{4},%
107             {sentencelong}{10},{longsentence}{10},{SentenceL}{10},%
108             {sentenceshort}{20},{shortsentence}{20},{SentenceS}{20},%
109             {sentencenumeric}{30},{numeralsentence}{30},{SentenceN}{30},%
110             {sentenceoff}{40},{nosentence}{40},%
111             {long}{11},%
112             {short}{22},%
113             {numeric}{33},%
114             {paragraphonly}{44},{onlyparagraph}{44},%
115             {ParagraphOnly}{44},{OnlyParagraph}{44}%
116         }{#1}%
117     \ifx\FamilyKeyState\FamilyKeyStateProcessed
118         \@tempcnta=\@tempa\relax
119         \@tempcntb=\z@
120         \@whilenum \@tempcnta>9 \do{%
121             \advance\@tempcnta -10\relax
122             \advance\@tempcntb \@ne\relax
123         }%
124     \ifcase \@tempcnta
125     \or

```

```

126     \aftergroup\def\aftergroup\parcite@format
127     \aftergroup{\aftergroup0\aftergroup}%
128   \or
129     \aftergroup\def\aftergroup\parcite@format
130     \aftergroup{\aftergroup1\aftergroup}%
131   \or
132     \aftergroup\def\aftergroup\parcite@format
133     \aftergroup{\aftergroup2\aftergroup}%
134   \or
135     \aftergroup\def\aftergroup\parcite@format
136     \aftergroup{\aftergroup-\aftergroup1\aftergroup}%
137   \fi
138   \ifcase \@tempcntb
139   \or
140     \aftergroup\def\aftergroup\sentencecite@format
141     \aftergroup{\aftergroup0\aftergroup}%
142   \or
143     \aftergroup\def\aftergroup\sentencecite@format
144     \aftergroup{\aftergroup1\aftergroup}%
145   \or
146     \aftergroup\def\aftergroup\sentencecite@format
147     \aftergroup{\aftergroup2\aftergroup}%
148   \or
149     \aftergroup\def\aftergroup\sentencecite@format
150     \aftergroup{\aftergroup-\aftergroup1\aftergroup}%
151   \fi
152   \fi
153   \endgroup
154 }
155 \KOMADeclareDeprecatedOption[scrjura]{parcitenam}{ref=parlong}
156 \KOMADeclareDeprecatedOption[scrjura]{sentencecitenam}{ref=sentencelong}

```

Optionen ausführen.

```
157 \KOMAProcessOptions\relax
```

`\jura@env@type` Dieses Makro wird in Abhängigkeit der gerade aktiven `scrjura`-Umgebung definiert.

```
158 \newcommand*{\jura@env@type}{}

```

`\ellipsispar` Paragraphen auslassen aber mit zählen.

`\parellipsis`

```

159 \newcommand*{\ellipsispar}[1][1]{%
160   \begingroup
161     \KOMAOptions{parnumber=manual}\parellipsis\par
162     \addtocounter{par}{#1}%
163     \protected@write\@auxout{}{

```

```

164     \string\newmaxpar{\jura@env@type}%
165     {\thecontractAbsoluteParagraph}%
166     {\thepar}%
167   }%
168 \endgroup
169 \addtocounter{par}{-1}\refstepcounter{par}%
170 \ignorespaces
171 }
172 \newcommand*{\parellipsis}{%
173   \scr@ifundefinedorrelax{textellipsis}{\dots}{\textellipsis}%
174 }

```

<pre> contract \contract contractParagraph \thecontractParagraph \contract@Paragraphformat \Paragraphformat contractSubParagraph \thecontractSubParagraph contractAbsoluteParagraph </pre>	<p>Die Umgebung <code>contract</code> wird als Anweisung <code>\contract</code> definiert. Nichts desto trotz handelt es sich dabei um eine Umgebung und sollte sie immer als Umgebung verwendet werden! Die Umgebung kann nicht geschachtelt werden. Sie darf aber enden und dann erneut verwendet werden. Die Paragraphen werden jedoch über alle Verträge hinweg nummeriert. Das Ende eines Vertrags ist also genau genommen nur eine Unterbrechung.</p> <pre> 175 \newcommand*{\contract}{% 176 \ifx\jura@env@type\@empty 177 \let\@doendpe\contract@doendpe 178 \let\Paragraph\contract@paragraph 179 \let\c@Paragraph\c@contractParagraph 180 \edef\cl@Paragraph{\cl@Paragraph\cl@contractParagraph}% 181 \let\SubParagraph\contract@subparagraph 182 \let\c@SubParagraph\c@contractSubParagraph 183 \edef\cl@SubParagraph{\cl@SubParagraph\cl@contractSubParagraph}% 184 \let\Par\contract@everypar 185 \let\Sentence\contract@sentence 186 \renewcommand*{\jura@env@type}{contract}% 187 \aliaskomafont{Paragraph}{contract.Paragraph}% 188 \else 189 \PackageError{contract}{nested ‘contract’ detected}{% 190 You may not use a ‘contract’ environment inside\MessageBreak 191 a ‘\jura@env@type’ environment or after loading\MessageBreak 192 package ‘scrjura’ with option ‘\jura@env@type’!}% 193 \fi 194 } 195 \let\if@jura@skiphyperref\iftrue 196 \let\cl@Paragraph\@empty 197 \let\cl@SubParagraph\@empty 198 \newcounter{contractParagraph} 199 \renewcommand*{\thecontractParagraph}{% 200 {\contract@Paragraphformat{\arabic{contractParagraph}}}} 201 \DeclareRobustCommand*{\contract@Paragraphformat}[1]{\Paragraphformat{#1}} 202 \newcommand*{\Paragraphformat}[1]{\S~#1} 203 \newcounter{contractSubParagraph} </pre>
--	--

```

204 \@addtoreset{SubParagraph}{Paragraph}
205 \renewcommand*{\thecontractSubParagraph}{%
206   {\theParagraph\alph{contractSubParagraph}}}
207 \newcounter{contractAbsoluteParagraph}

```

`\contract@paragraph` Das ist das Macro, das in Verträgen `\Paragraph` zur Verfügung stellt. Ein Vertrag besteht aus mehreren Paragraphen. Jeder Paragraph hat einige optionale Elemente, die (auf dem Umweg über `scrkbase`) über das `keyval`-Paket geregelt werden.

```

title Der Titel, der Kolumnentitel und der Verzeichniseintrag des Paragraphen.
head Der Titel setzt dabei zunächst auch die anderen beiden. Man kann die
entry beiden aber auf leer setzen. Insbesondere aber kann man sie mit den no-
tocentry Optionen auch abschalten.
nohead
noentry
notocentry
208 \define@key{contract}{title}{%
209   \def\contract@title{#1}%
210   \ifx\contract@entry\relax\def\contract@entry{\contract@title}\fi
211   \ifx\contract@head\relax\def\contract@head{\contract@title}\fi
212 }
213 \define@key{contract}{entry}{%
214   \PackageWarning{scrjura}{deprecated option 'entry'.\MessageBreak
215     You should use option 'tocentry' instead of\MessageBreak
216     option 'entry'%
217   }%
218   \def\contract@entry{#1}}
219 \define@key{contract}{tocentry}{\def\contract@entry{#1}}
220 \define@key{contract}{noentry}[]{%
221   \PackageWarning{scrjura}{deprecated option 'noentry'.\MessageBreak
222     You should use option 'notocentry' instead of\MessageBreak
223     option 'noentry'%
224   }%
225   \let\contract@entry\relax}
226 \define@key{contract}{notocentry}[]{\let\contract@entry\relax}
227 \define@key{contract}{head}{\def\contract@head{#1}}
228 \define@key{contract}{nohead}[]{\let\contract@head\relax}

```

`number` Die Nummer kann mit dieser Option frei gestaltet werden. Es sind jedoch keine Paragraphen ohne Nummer erlaubt. Wird keine Nummer angegeben, so wird die Nummer automatisch gesetzt.

```

229 \define@key{contract}{number}{\def\contract@number{#1}}

```

`\contract@preskip` Über diese beiden Optionen kann der Abstand vor und nach einem Paragraphen gesetzt werden. Wird nichts angegeben, so wird die globale Voreinstellung verwendet, die übrigens mit `\setkeys{contract}{...}` gesetzt werden kann.

`\contract@postskip`

`preskip`

`postskip`

```

230 \newcommand*{\contract@preskip}{2\baselineskip}
231 \newcommand*{\contract@postskip}{\baselineskip}
232 \define@key{contract}{preskip}{\def\contract@preskip{#1}}
233 \define@key{contract}{postskip}{\def\contract@postskip{#1}}

```

dummy Über dies Option wird der Schalter `\ifcontract@dummy` auf `\iftrue` gesetzt.

`\ifcontract@dummy` Ist der Schalter `\iftrue`, dann wird der Paragraph nicht gesetzt. Es ist jedoch darauf zu achten, dass Absätze und Sätze des Paragraphen auf diese Weise nicht entfernt werden können. Werden jedoch Paragraphen gelöscht, so kann man damit das Loch in der Nummerierung erzeugen.

```

234 \newif\ifcontract@dummy
235 \define@key{contract}{dummy}[true]{\csname contract@dummy#1\endcsname}

```

`\contract@paragraph@font` Das muss noch auf die Element-Schnittstelle von KOMA-Script umgestellt werden!

```

236 \newkomafont{contract.Paragraph}{\sffamily\bfseries\large}
237 \newcommand*{\contract@paragraph@font}{\usekomafont{contract.Paragraph}%
238 \@hangfrom}

```

```

@AbsParagraph
\theH@AbsParagraph
\theHParagraph
\theHSubParagraph
239 % \textsf{hyperref}-Code, der noch nicht getestet ist!
240 \newcounter{@AbsParagraph}
241 \newcommand*{\theH@AbsParagraph}[P-\arabic{@AbsParagraph}]
242 \newcommand*{\theHParagraph}{\theH@AbsParagraph}
243 \newcommand*{\theHSubParagraph}{\theH@AbsParagraph}

```

Zunächst wird vorübergehend auf manuelle Absatznummern umgeschaltet, weil natürlich innerhalb der Überschrift keine Nummer gesetzt werden soll. Trotzdem werden generell die Absatznummern eingeschaltet. Dann werden die Optionen initialisiert und ausgeführt.

```

244 \newcommand*{\contract@paragraph}[1]{%
245 \stepcounter{contractAbsoluteParagraph}%
246 \ManualPar\parnumbertrue
247 \let\contract@title\relax
248 \let\contract@entry\relax
249 \let\contract@head\relax
250 \let\contract@number\relax
251 \contract@dummyfalse
252 \ifx\relax#1\relax\else\setkeys{contract}{#1}\fi

```

Wenn es kein Blindparagraph ist, werden Vorbereitung für nach der Überschrift getroffen und der vertikale Abstand eingefügt.

```

253 \ifcontract@dummy\else
254   \par
255   \@afterindentfalse
256   \addvspace{\contract@preskip}%
257 \fi

```

Wenn die Nummer nicht manuell gesetzt wurde, wird die nächste Nummer verwendet. Im anderen Fall muss die Nummer gesetzt und auch dafür gesorgt werden, dass sowohl ein Label als auch hyperref diese Nummer verwenden.

```

258 \ifx\contract@number\relax
259   \let\p@Paragraph\@empty
260   \let\theParagraph\thecontractParagraph
261   \refstepcounter{Paragraph}%
262 \else
263   \begingroup
264     \let\@elt\@stpelt
265     \cl@Paragraph
266   \endgroup
267   \protected@edef\theParagraph{\contract@Paragraphformat{\contract@number}}%
268   \protected@edef\@currentlabel{\theParagraph}%
269 \fi
270 \stepcounter{@AbsParagraph}%
271 \begingroup\expandafter\expandafter\expandafter\endgroup
272 \expandafter\ifx\csname if@skiphyperref\endcsname\relax
273 \else
274   \expandafter\let\csname if@jura@skiphyperref\endcsname
275   \csname if@skiphyperref\endcsname
276 \fi
277 \if@jura@skiphyperref\else
278   \hyper@refstepcounter{@AbsParagraph}%
279   \typeout{absolute Nummer: \the@AbsParagraph^^JLabel: '\@currentHref'}%
280 \fi

```

Zwecks Vereinfachung wird ab hier so getan, als wäre es ein Unterabschnitt.

```

281 \let\theSubParagraph\theParagraph

```

Außer für Blindparagraphen wird dann die Überschrift gesetzt, die Verzeichniseinträge vorgenommen und der Kolumnentitel angepasst.

```

282 \ifcontract@dummy\else
283   \begingroup
284     \if@juratitlepagebreak\else\interlinepenalty\@M\fi
285     \contract@paragraph@font{\theParagraph
286       \ifx\contract@title\relax\else\enskip\fi}%

```

```

287     \contract@title
288     \ifx\contract@entry\relax\else
289         \addxcontentsline{toc}{cpar}[\theParagraph]\contract@entry
290         \addxcontentsline{cpa}{cpar}[\theParagraph]\contract@entry
291     \fi
292     \ifx\contract@head\relax\else
293         \expandafter\Paragraphmark\expandafter{%
294             \expandafter\theSubParagraph\expandafter\enskip\contract@head}%
295     \fi
296     \par
297 \endgroup\nobreak\vskip\contract@postskip

```

Zum Schluss wird noch dafür gesorgt, dass auch die Absätze korrekt nummeriert werden können etc.

```

298     \jura@afterheading{contract}%
299 \fi
300 }

```

`\contract@subparagraph` Das entspricht bis auf wenige Kleinigkeiten, die Unterparagrafen betreffen, `\contract@paragraph`

```

301 \newcommand*{\contract@subparagraph}[1]{%
302     \stepcounter{contractAbsoluteParagraph}%
303     \ManualPar\parnumbertrue
304     \let\contract@title\relax
305     \let\contract@entry\relax
306     \let\contract@head\relax
307     \let\contract@number\relax
308     \contract@dummysfalse
309     \ifx\relax#1\relax\else\setkeys{contract}{#1}\fi
310     \ifcontract@dummys\else
311         \par
312         \@afterindentfalse
313         \vskip\contract@preskip
314     \fi
315     \ifx\contract@number\relax
316         \let\p@SubParagraph\@empty
317         \let\theSubParagraph\thecontractSubParagraph
318         \refstepcounter{SubParagraph}%
319     \else
320         \begingroup
321             \let\@elt\@stpelt
322             \cl@SubParagraph
323         \endgroup
324         \protected@edef\theSubParagraph{\theParagraph\contract@number}%
325         \protected@edef\@currentlabel{\theSubParagraph}%
326     \fi
327     \stepcounter{@AbsParagraph}%

```

```

328 \begingroup\expandafter\expandafter\expandafter\endgroup
329 \expandafter\ifx\csname if@skiphyperref\endcsname\relax
330 \else
331   \expandafter\let\csname if@jura@skiphyperref\expandafter\endcsname
332   \csname if@skiphyperref\endcsname
333 \fi
334 \if@jura@skiphyperref\else
335   \hyper@refstepcounter{@AbsParagraph}%
336   \typeout{absolute Nummer: \the@AbsParagraph^^JLabel: '\@currentHref'}%
337 \fi
338 \ifcontract@dummys\else
339   \begingroup
340     \if@juratitlepagebreak\else\interlinepenalty\@M\fi
341     \contract@paragraph@font{\theSubParagraph
342       \ifx\contract@title\relax\else\enskip\fi}%
343     \contract@title
344     \ifx\contract@entry\relax\else
345       \addxcontentsline{toc}{cpar}[\theSubParagraph]\contract@entry
346       \addxcontentsline{cpa}{cpar}[\theSubParagraph]\contract@entry
347     \fi
348     \ifx\contract@head\relax\else
349       \expandafter\Paragraphmark\expandafter{%
350         \expandafter\theSubParagraph\expandafter\enskip\contract@head}%
351     \fi
352     \par
353   \endgroup
354   \nobreak\vskip\contract@postskip
355   \jura@afterheading{contract}%
356 \fi
357 }

```

`\AutoPar` Automatische oder manuelle Absatznummern für alle Umgebungen aktivieren.
`\ManualPar`

```

358 \newcommand*{\AutoPar}{%
359   \expandafter\let\expandafter\jura@everypar
360   \csname \jura@type @everypar\endcsname
361 }
362 \newcommand*{\ManualPar}{%
363   \let\jura@everypar\relax
364 }

```

`\jura@afterheading` Entspricht `\afterheading` mit Erweiterungen für automatischen Absatznummern.

```

365 \newcommand*{\jura@afterheading}[1]{%
366   \@nobreaktrue
367   \everypar{%

```

```

368 \if@nobreak
369 \@nobreakfalse
370 \clubpenalty \@M
371 \if@afterindent \else
372 {\setbox\z@\lastbox}%
373 \fi
374 \else
375 \clubpenalty \@clubpenalty
376 \everypar{%
377 \jura@everypar
378 }%
379 \fi
380 \jura@everypar
381 }%
382 \def\jura@type{#1}%
383 \AutoPar
384 }

```

`\jura@everypar` Die Anweisung, die am Anfang von jedem Absatz auszuführen ist, um die Nummer zu setzen. Außerhalb der Umgebungen ist das nichts.

```
385 \newcommand*\jura@everypar{}
```

`\@doendpe` Diese Anweisung wird von L^AT_EX verwendet, um am Ende von Umgebungen die kurz-, mittel- und langwirkenden Absatzaktionen zurückzusetzen. Damit hier nicht die automatische Absatznummerierung abgeschaltet wird, wird sie neu eingefügt.

```

386 \CheckCommand*\@doendpe{\@endpetrue
387 \def\par{\@restorepar\everypar}\par\@endpefalse}\everypar
388 {\setbox\z@\lastbox}\everypar{\@endpefalse}}
389 \newcommand*\contract@doendpe{%
390 \@endpetrue
391 \def\par{%
392 \@restorepar\everypar{%
393 \csname jura@everypar\endcsname
394 }%
395 \par\@endpefalse
396 }%
397 \everypar{%
398 {\setbox\z@\lastbox}\everypar{%
399 \csname jura@everypar\endcsname
400 }%
401 \@endpefalse
402 }%
403 }

```

`\l@cpa`r Verzeichnis-Eintrag für einen Vertrags-Paragraphen.

```
404 \newcommand*{\l@cpa}{2}{%
405   \ifnum\value{juratoclevel}>\value{tocdepth}\else
406     \scr@ifundefinedorrelax{bprot@dottedtocline}{%
407       \@dottedtocline
408     }{%
409       \bprot@dottedtocline
410     }{\value{juratoclevel}}{\cpar@indent}{\cpar@numberwidth}{#1}{#2}%
411   \fi
412 }
```

`\jura@separator` Wird verwendet, damit Leerzeichen am Anfang oder Ende entfernt werden können.

```
413 \DeclareRobustCommand*\jura@separator[1]{#1}
```

`\jura@usetype` In der Voreinstellung macht das erst einmal gar nichts, außer dass es robust ist und `\jura@@usetype` aufruft.

`\jura@@usetype` Das wiederum ist nicht robust und kann dadurch einfacher undefiniert werden. In der Voreinstellung macht das dann wirklich nichts, außer das Argument zu fressen.

```
414 \DeclareRobustCommand*\jura@usetype[1]{\jura@@usetype{#1}}
415 \newcommand*\jura@@usetype[1]{}
```

`\contract@everypa`r `\jura@everypa`r für Verträge.

`\ifparnumber` Der Schalter gibt an, ob überhaupt mit Absatznummern gearbeitet werden soll. Sind Absatznummern deaktiviert, werden auch keine manuellen
`par` Absatznummern gesetzt. Die Absätze werden dann auch nicht gezählt. An-
`\thepa`r sonst werden die Absätze mit `\thepa`r nummeriert. Wichtig ist, dass der
`\theHpa`r Absatzzähler mit den Paragraphen und den Unterparagraphen zurückge-
`\parformat` setzt wird. Außerdem muss beim Zitieren als Elternobjekt der Paragraph
`\p@pa`r mit ausgegeben werden.

```
416 \newcounter{par}
417 \renewcommand*\thepa{\arabic{par}}
418 \newcommand*\theHpa{\theH@AbsParagraph-\Roman{par}}
419 \newcommand*\parformat{(\thepa)}
420 \newkomafont{parnumber}{}
421 \renewcommand*\p@pa{\jura@usetype{\jura@env@type}\theSubParagraph\jura@separator}
422 \@addtoreset{par}{Paragraph}
423 \@addtoreset{par}{SubParagraph}
```

```

424 \newcommand*{\contract@everypar}{%
425   \ifparnumber
426     \ifx\scrjura@special@par\relax
427       \ifx\scrjura@special@reset@par\relax\else
428         \global\let\thepar\scrjura@special@reset@par
429         \global\let\scrjura@special@reset@par\relax
430       \fi
431       \refstepcounter{par}%
432       \refstepcounter{sentence}%
433     \else
434       \ifx\scrjura@special@reset@par\relax
435         \global\let\scrjura@special@reset@par\thepar
436       \fi
437       \global\let\thepar\scrjura@special@par
438       \global\let\scrjura@special@par\relax
439       \setcounter{sentence}{0}\refstepcounter{sentence}%
440     \fi
441   \begingroup
442     \protected@write\auxout{\let\contract@Paragraphformat\@firstofone}{%
443       \string\newmaxpar{contract}%
444         {\thecontractAbsoluteParagraph}%
445         {\thepar}}%
446     \getmaxpar\@tempa{contract}{\thecontractAbsoluteParagraph}%
447     \typeout{Stored max is \@tempa}%
448     \def\reserved@a##1\@nnil{\def\@tempa{##1}}%
449     \afterassignment\reserved@a\@tempcnta=0\@tempa\relax\@nnil
450     \ifnum \@tempcnta>\@ne
451       {\usekomafont{parnumber}{\parformat\nobreakspace}}%
452     \else
453       \def\reserved@a{\relax}%
454       \ifx\@tempa\reserved@a
455         \else
456           {\usekomafont{parnumber}{\parformat\nobreakspace}}%
457         \fi
458       \fi
459     \endgroup
460   \else
461     \setcounter{sentence}{-1}\refstepcounter{sentence}%
462   \fi
463 }

```

`\thisparnumber` Mit dieser Anweisung kann man eine Absatznummer manuell vergeben.
`\scrjura@special@par` Allerdings muss die Nummer voll expandierbar sein!
`\scrjura@special@reset@par`

```

464 \newcommand*{\thisparnumber}[1]{%
465   \def\scrjura@special@par{#1}%
466 }
467 \newcommand*{\scrjura@special@par}{}
468 \let\scrjura@special@par\relax

```

```

469 \newcommand*\scrjura@special@reset@par{}
470 \let\scrjura@special@reset@par\relax

```

`\refL` Das gleiche wie `\ref` aber zwingend mit der Langform.

```

\ref@L
471 \newcommand*\refL{\@ifstar {\ref@L*}{\ref@L{}}}
472 \newcommand*\ref@L}[2]{%
473   \begingroup
474     \def\parcite@format{0}%
475     \let\sentencecite@format\parcite@format
476     \ref#1{#2}%
477   \endgroup
478 }

```

`\refS` Das gleiche wie `\ref` aber zwingend mit der Kurzform.

```

\ref@S
479 \newcommand*\refS{\@ifstar {\ref@S*}{\ref@S{}}}
480 \newcommand*\ref@S}[2]{%
481   \begingroup
482     \def\parcite@format{1}%
483     \let\sentencecite@format\parcite@format
484     \ref#1{#2}%
485   \endgroup
486 }

```

`\refN` Das gleiche wie `\ref` aber zwingend mit der numerischen Form.

```

\ref@N
487 \newcommand*\refN{\@ifstar {\ref@N*}{\ref@N{}}}
488 \newcommand*\ref@N}[2]{%
489   \begingroup
490     \def\parcite@format{2}%
491     \let\sentencecite@format\parcite@format
492     \ref#1{#2}%
493   \endgroup
494 }

```

`\refParagraph` Zitiert nur den Paragraphen eines Paragraphen, eines Absatzes oder eines Satzes. Für mehr Kompatibilität mit `hyperref` gibt es mit `hyperref` auch eine Sternform. Ohne `hyperref` produziert diese nur Unsinn.

```

495 \newcommand*\refParagraph{%
496   \@ifstar {\ref@Paragraph*}{\ref@Paragraph{}}
497 }
498 \newcommand*\ref@Paragraph}[2]{%
499   \expandafter\ifx\csname r@#2\endcsname\relax
500     \ref#1{#2}%
501   \else
502     \begingroup

```

Alle Teile der Referenz bis auf den ersten in \@tempb ablegen.

```
503     \expandafter\expandafter\expandafter\expandafter
504     \expandafter\expandafter\expandafter\def
505     \expandafter\expandafter\expandafter\expandafter
506     \expandafter\expandafter\expandafter\@tempb
507     \expandafter\expandafter\expandafter\expandafter
508     \expandafter\expandafter\expandafter{%
509     \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%
```

Den ersten Teil der Referenz in \@tempa ablegen.

```
510     \def\@tempc##1##2\@nil{##1}%
511     \let\jura@separator\@gobble
512     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
513     \csname r@#2\endcsname\noexpand\@nil}%
```

Den ersten Teil von \@tempa in \@tempa ablegen.

```
514     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
515     \@tempa\@nil}%
516     \let\@@protect\protect
517     \let\protect\noexpand
518     \expandafter\edef\csname r@#2\endcsname{{\@tempa}\@tempb}%
519     \let\protect\@@protect
520     \ref#1{#2}%
521 \endgroup
522 \fi
523 }
```

`\refParagraphN` Zitiert nur den Paragraphen eines Paragraphen, eines Absatzes oder eines
`\ref@ParagraphN` Satzes. Für mehr Kompatibilität mit hyperref gibt es mit hyperref auch
eine Sternform. Ohne hyperref produziert diese nur Unsinn.

```
524 \newcommand*{\refParagraphN}{%
525   \@ifstar {\ref@ParagraphN*}{\ref@ParagraphN{}}
526 }
527 \newcommand*{\ref@ParagraphN}[2]{%
528   \begingroup
529     \let\Paragraphformat\relax
530     \ref@Paragraph{#1}{#2}%
531   \endgroup
532 }
```

`\refPar` Zitiert nur den Absatz eines Absatzes oder eines Satzes. Für mehr Kom-
`\ref@Par` patibilität mit hyperref gibt es mit hyperref auch eine Sternform. Ohne
hyperref produziert diese nur Unsinn.

```
533 \newcommand*{\refPar}{%
```

```

534 \@ifstar {\ref@Par*}{\ref@Par{}}
535 }
536 \newcommand*{\ref@Par}[2]{%
537 \expandafter\ifx\csname r@#2\endcsname\relax
538 \ref#1{#2}%
539 \else
540 \begingroup

```

Alle Teile der Referenz bis auf den ersten in \@tempb ablegen.

```

541 \expandafter\expandafter\expandafter\expandafter
542 \expandafter\expandafter\expandafter\def
543 \expandafter\expandafter\expandafter\expandafter
544 \expandafter\expandafter\expandafter\@tempb
545 \expandafter\expandafter\expandafter\expandafter
546 \expandafter\expandafter\expandafter{%
547 \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%

```

Den ersten Teil der Referenz in \@tempa ablegen.

```

548 \def\@tempc##1##2\@nil{##1}%
549 \let\jura@separator\@gobble
550 \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
551 \csname r@#2\endcsname\noexpand\@nil}%

```

Den zweiten Teil von \@tempa in \@tempa ablegen.

```

552 \def\@tempc##1##2##3\@nil{##2}%
553 \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
554 \@tempa{%
555 \protect\G@refundefinedtrue
556 \nfss@text{\reset@font\bfseries ??}%
557 \@latex@warning{Reference ‘#2’ on page \thepage \space
558 with undefined par number}%
559 }\noexpand\@nil}%
560 \let\@@protect\protect
561 \let\protect\noexpand
562 \expandafter\edef\csname r@#2\endcsname{\@tempa}\@tempb}%
563 \let\protect\@@protect
564 \ref#1{#2}%
565 \endgroup
566 \fi
567 }

```

```

\refParL
\ref@ParX

```

```

568 % Das Gleiche zwingend lang.
569 \newcommand*{\refParL}{%
570 \@ifstar {\ref@ParX*}{\ref@ParX{}}
571 }

```

```

572 \newcommand*{\ref@ParX}[3]{%
573   \begingroup
574     \def\parcite@format{#1}%
575     \let\sentencecite@format\parcite@format
576     \ref@Par{#2}{#3}%
577   \endgroup
578 }

```

`\refParS` Das Gleiche zwingend kurz.

```

579 \newcommand*{\refParS}{%
580   \@ifstar {\ref@ParX1*}{\ref@ParX1{}}
581 }

```

`\refParN` Das Gleiche zwingend numerisch.

```

582 \newcommand*{\refParN}{%
583   \@ifstar {\ref@ParN2*}{\ref@ParN2{}}
584 }
585 \newcommand*{\ref@ParN}[2]{%
586   \@ifnextchar [%]
587     {\ref@@ParN{#1}{#2}}%
588     {\ref@ParX{#1}{#2}}%
589 }
590 \newcommand*{\ref@@ParN}{%
591   \def\ref@@ParN#1#2[#3]#4{%
592     \begingroup
593       \renewcommand*{\parnumericformat}[1]{%
594         \csname @#3\endcsname{\number ##1\relax}%
595       }%
596       \ref@ParX{#1}{#2}{#4}%
597     \endgroup
598 }

```

`\refSentence` Zitiert nur den Satz eines Satzes. Für mehr Kompatibilität mit hyperref
`\ref@Sentence` gibt es mit hyperref auch eine Sternform. Ohne hyperref produziert diese
 nur Unsinn.

```

599 \newcommand*{\refSentence}{%
600   \@ifstar {\ref@Sentence*}{\ref@Sentence{}}
601 }
602 \newcommand*{\ref@Sentence}[2]{%
603   \expandafter\ifx\csname r#2\endcsname\relax
604     \ref#1{#2}%
605   \else
606     \begingroup

```

Alle Teile der Referenz bis auf den ersten in \@tempb ablegen.

```
607 \expandafter\expandafter\expandafter\expandafter
608 \expandafter\expandafter\expandafter\def
609 \expandafter\expandafter\expandafter\expandafter
610 \expandafter\expandafter\expandafter\@tempb
611 \expandafter\expandafter\expandafter\expandafter
612 \expandafter\expandafter\expandafter{%
613 \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%
```

Den ersten Teil der Referenz in \@tempa ablegen.

```
614 \def\@tempc##1##2\@nil{##1}%
615 \let\jura@separator\@gobble
616 \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
617 \csname r@#2\endcsname\noexpand\@nil}%
```

Den dritten Teil von \@tempa in \@tempa ablegen.

```
618 \def\@tempc##1##2##3##4\@nil{##3}%
619 \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
620 \@tempa}{%
621 \protect\G@refundefinedtrue
622 \nfss@text{\reset@font\bfseries ??}%
623 \@latex@warning{Reference ‘#2’ on page \thepage \space
624 with undefined sentence number}%
625 }\noexpand\@nil}%
626 \let\@@protect\protect
627 \let\protect\noexpand
628 \expandafter\edef\csname r@#2\endcsname{{\@tempa}\@tempb}%
629 \let\protect\@@protect
630 \ref#1{#2}%
631 \endgroup
632 \fi
633 }
```

\refSentenceL

\ref@SentenceX

```
634 % Das Gleiche zwingend lang.
635 \newcommand*{\refSentenceL}{%
636 \@ifstar {\ref@SentenceX0*}{\ref@SentenceX0{}}
637 }
638 \newcommand*{\ref@SentenceX}[3]{%
639 \begingroup
640 \def\parcite@format{#1}%
641 \let\sentencecite@format\parcite@format
642 \ref@Sentence{#2}{#3}%
643 \endgroup
644 }
```

`\refSentenceS` Das Gleiche zwingend kurz.

```
645 \newcommand*{\refSentenceS}{%
646   \@ifstar {\ref@SentenceX1*}{\ref@SentenceX1{}}
647 }
```

`\refSentenceN` Das Gleiche zwingend numerisch.

```
648 \newcommand*{\refSentenceN}{%
649   \@ifstar {\ref@SentenceX2*}{\ref@SentenceX2{}}
650 }
```

`\contract@sentence` Sätze nummerieren.

`sentence` Der Zähler wird für die Nummerierung der Sätze verwendet. Dabei ist
`\thesentence` wichtig, dass beim Zitieren der Sätze als Elternobjekt der Absatz mit aus-
`\theHsentence` gegeben wird.

```
\p@sentence
651 \newcounter{sentence}[par]
652 \renewcommand*{\thesentence}{\arabic{sentence}}
653 \newcommand*{\theHsentence}{\theHpar-\arabic{sentence}}
654 \renewcommand*{\p@sentence}{\expandafter\p@@sentence}
655 \newcommand*{\p@@sentence}[1]{\p@par{\par@cite{\thepar}}%
656   \jura@separator{\nobreakspace}}{\sentence@cite{#1}}
657 \newcommand*{\contract@sentence}{%
658   \refstepcounter{sentence}%
659   \textsuperscript{\thesentence}\nobreak\hskip\z@
660 }
```

`\parciteformat` Zitierstil für Absätze.

```
661 \DeclareRobustCommand*{\par@cite}[1]{\parciteformat{#1}}
662 \newcommand*{\parciteformat}[1]{%
663   \ifcase \parcite@format
664     \expandafter\parlongformat
665   \or
666     \expandafter\parshortformat
667   \or
668     \expandafter\parnumericformat
669   \else
670     \unskip\expandafter\@gobble
671   \fi
672   {#1}%
673 }
```

`\sentenceciteformat` Zitierstil für Sätze. In der Voreinstellung ist das `\@arabic`.

```
674 \DeclareRobustCommand*\sentence@cite}[1]{\sentenceciteformat{#1}}
675 \newcommand*\sentenceciteformat}[1]{%
676   \ifcase \sentencecite@format
677     \expandafter\sentencelongformat
678   \or
679     \expandafter\sentenceshortformat
680   \or
681     \expandafter\sentencenumericformat
682   \else
683     \unskip\expandafter\@gobble
684   \fi
685   {#1}%
686 }
```

`\parlongformat` Formatierung in den sechs Formen.

```
\parshortformat
\parnumericformat
\sentencelongformat
\sentenceshortformat
\sentencenumericformat
687 \newcommand*\parlongformat}[1]{\parname~#1}
688 \newcommand*\parshortformat}[1]{\parshortname~#1}
689 \newcommand*\parnumericformat}[1]{\@Roman{\number #1\relax}}
690 \newcommand*\sentencelongformat}[1]{\sentencename~#1}
691 \newcommand*\sentenceshortformat}[1]{\sentenceshortname~#1}
692 \newcommand*\sentencenumericformat}[1]{\@arabic{\number #1\relax}.}
```

`\parname` Der Name eines Absatzes und eines Satzes. Die englischen Namen wurden

`\parshortname` von »m.eik« beigesteuert.

`\sentencename`

`\sentenceshortname` 693 \newcommand*\parname}{Paragraph}

```
694 \AtBeginDocument{%
695   \providecaptionname{german,ngerman,austrian,naustrian}\parname{Absatz}%
696   \providecaptionname{german,ngerman,austrian,naustrian}\parshortname{Abs.}%
697   \providecaptionname{german,ngerman,austrian,naustrian}\sentencename{Satz}%
698   \providecaptionname{german,ngerman,austrian,naustrian}\sentenceshortname{S.}%
699   \providecaptionname{english,american,british,canadian,USenglish,UKenglish}\parname{Paragraph}%
700   \providecaptionname{english,american,british,canadian,USenglish,UKenglish}\parshortname{Par.}%
701   \providecaptionname{english,american,british,canadian,USenglish,UKenglish}\sentencename{Sentence}%
702   \providecaptionname{english,american,british,canadian,USenglish,UKenglish}\sentenceshortname{S.}%
703 }
704 \providecommand*\parname}{\scrjura@lang@error{\parname}}
705 \providecommand*\parshortname}{\scrjura@lang@error{\parshortname}}
706 \providecommand*\sentencename}{\scrjura@lang@error{\sentencename}}
707 \providecommand*\sentenceshortname}{\scrjura@lang@error{\sentenceshortname}}
708 \newcommand*\scrjura@lang@error}[1]{%
709   \PackageError{scrjura}{%
710     current language not supported%
711   }{%
712     Currently scrjura only supports languages ‘german’, ‘ngerman’,
713     ‘austrian’,\MessageBreak
```

```

714     'naustrian', 'english', 'american', 'british', 'canadian', 'USenglish',
715     and\MessageBreak
716     'UKenglish'.\MessageBreak
717     It seems, that you are using another language (maybe '\language') or
718     that\MessageBreak
719     your language selection isn't compatible to package 'babel'.\MessageBreak
720     Because of this you have to define '\string#1' by yourself!\MessageBreak
721     It would be nice if you'll send your definitions to the author.%
722 }%
723 \textbf{??}%
724 }

```

`\newmaxpar` Noch zwei Hilfsmakros, um Zähler in der aux-Datei zwischenzuspeichern
`\getmaxpar` und auch dann Werte zu holen, wenn sie nicht in der aux-Datei stehen.

```

725 \newcommand*\newmaxpar}[3]{%
726   \begingroup
727   \expandafter\let\csname #1@Paragraphformat\endcsname\@firstofone
728   \protected@edef\@tempa{#2}\@onelevel@sanitize\@tempa
729   \expandafter\xdef\csname max@#1@\@tempa\endcsname{#3}%
730 \endgroup
731 }
732 \newcommand*\getmaxpar}[3]{%
733   \begingroup
734   \expandafter\let\csname #2@Paragraphformat\endcsname\@firstofone
735   \protected@edef\@tempa{#3}%
736   \@onelevel@sanitize\@tempa
737   \expandafter\ifx \csname max@#2@\@tempa\endcsname\relax
738     \edef\@tempa{\endgroup\edef\noexpand#1{\expandafter\the\value{par}}}%
739   \else
740     \edef\@tempa{\endgroup
741       \edef\noexpand#1{\csname max@#2@\@tempa\endcsname}}%
742   \fi
743   \@tempa
744 }

```

Index

Numbers written in *italic* refer to the page where the corresponding entry is described; numbers underlined refer to the code line of the definition; numbers in **roman** refer to the code lines where the entry is used.

Symbols	A
<code>\AutoPar</code>	<i>10</i> , <u>358</u>
<code>@AbsParagraph</code> (Zähler)	<u>239</u>
	C
<code>\@doendpe</code>	<u>386</u>
<code>\contract</code>	<u>175</u>

paragraphmark (Option)	0, 53	\scrjura@special@par	464
\parcite@fromat	98	\scrjura@special@reset@par	464
\parciteformat	10, 661	\Sentence	9
parcitename (Option)	98	\sentence	10
\parellipsis	6, 159	sentence (Zähler)	651
\parformat	10, 416	\sentencecite@fromat	98
\parlongformat	687	\sentenceciteformat	10, 674
\parname	10, 693	sentencecitename (Option)	98
parnumber (Option)	45	\sentencelongformat	687
\parnumberfalse	9, 10	\sentencename	10, 693
\parnumbertrue	9, 10	\sentencenumericformat	687
\parnumericformat	687	\sentenceshortformat	687
\parshortformat	687	\sentenceshortname	693
\parshortname	693	\SubParagraph	6
postskip (Option)	0, 230		
preskip (Option)	0, 230		
		T	
		\thecontractParagraph	10, 175
		\thecontractSubParagraph	10, 175
		\theH@AbsParagraph	239
		\theHpar	416
		\theHParagraph	239
		\theHsentence	651
		\theHSubParagraph	239
		\thepar	10, 416
		\thesentence	10, 651
		\thisparnumber	464
		title (Option)	208
		tocentry (Option)	208
		\toclevel@cpar	15
		U	
		Umgebungen:	
		contract	175
		Z	
		Zähler:	
		@AbsParagraph	239
		contractAbsoluteParagraph	175
		contractParagraph	175
		contractSubParagraph	175
		juratoclevel	15
		par	416
		sentence	651
		S	
\scrjura@lang@error	693		

Change History

0.5c	<code>\contract@everypar</code> : <code>\contract@Paragraphformat</code> expandiert beim Schreiben zu seinem Argument 24	<code>\getmaxpar</code> : Argument 3 muss <code>\formatprotected@edef</code> expandiert werden 32
	erstes Argument von <code>\newmaxpar</code> und zweites Argument von <code>\getmaxpar</code> ist <code>contract</code> nicht <code>contractpars</code> 24	<code>\if@juratoc</code> : Schalter durch Zähler überflüssig 11
v0.5b	<code>juratitlepagebreak</code> : neue Option 12	<code>\ifparnumber</code> : Definition des Schalters in die Definition der Option verschoben 23
v0.5c	<code>\getmaxpar</code> : <code>\#2@Paragraphformat</code> expandiert zu seinem Argument 32	<code>entry</code> : Option veraltet 17
	<code>\protected@edef</code> durch <code>\edef</code> ersetzt 32	<code>tocentry</code> : neue Option 17
v0.5d	<code>\contract</code> : <code>\jura@env@type</code> wird gesetzt 16	<code>noentry</code> : Option veraltet 17
	fehlende Fehlermeldung ergänzt 16	<code>notocentry</code> : neue Option 17
	Umgebung definiert sich nicht mehr selbst um 16	<code>juratocindent</code> : neu 12
	<code>ref</code> : neue Einstellungen <code>nopar</code> , <code>nosentence</code> , <code>OnlyParagraph</code> 14	<code>juratoclevel</code> : Zählerdefinition verschoben 11
	<code>\jura@@usetype</code> : neu (intern) 23	<code>juratocnumberwidth</code> : neu . . . 12
	<code>\jura@env@type</code> : neu (intern) 15	v0.6a
	<code>\jura@usetype</code> : neu (intern) . 23	<code>parnumber</code> : Meldung bei falschen Werten korrigiert 12
	<code>\p@par</code> : <code>\jura@usetype</code> mit Argument <code>\jura@env@type</code> eingefügt 23	<code>\contract@paragraph</code> : fehlendes <code>\par</code> ergänzt 19
	<code>\parciteformat</code> : Argument verschoben 30	<code>\contract@ subparagraph</code> : fehlendes <code>\par</code> ergänzt . . . 20
	<code>\sentenceciteformat</code> : Argument verschoben 31	v0.6b
v0.5e	<code>\Paragraphmark</code> : Beachtung von <code>\MakeMarkcase</code> 13	<code>\contract@paragraph</code> : Verwendung von <code>\contract@Paragraphformat</code> auch bei manueller Nummerierung der Paragraphen. 19
v0.6	<code>parnumber</code> : neue Option 12	<code>\newmaxpar</code> : <code>\#1@Paragraphformat</code> expandiert zu seinem Argument 32
	<code>\cpar@indent</code> : neu (intern) . . 12	<code>\Paragraphformat</code> : <code>\paragraphformat</code> undefiniert 16
	<code>\cpar@numberwidth</code> : neu (intern) 12	v0.7
		<code>paragraphmark</code> : Verwendung der renovierten Schnittstelle mit <code>\FamilyKeyStateProcessed</code> 13
		<code>parnumber</code> : Verwendung der renovierten Schnittstelle mit <code>\FamilyKeyStateProcessed</code> 12
		<code>\contract@everypar</code> : Absatznummer von <code>\thisparnumber</code> beachtet 24

Im Fall von nicht rein numerischen Absatznummern die Absätze auf jeden Fall nummerieren	24	<code>\parformat:</code> neues Font-Element <code>parnumber</code>	23
Satznummer bei manuell nummerierten Absätzen auf 0 statt 1 initialisiert, weil <code>\Sentence</code> die Nummer als erstes hochzählt.	24	<code>\scrjura@lang@error:</code> Verwendung von <code>\PackageError</code> an Stelle von <code>\PackageErrorNoLine</code>	31
Zähler <code>contractAbsoluteParagraph</code> wird verwendet	24	<code>\scrjura@special@par:</code> neue Anweisung (intern)	24
<code>\contract@paragraph:</code> Abstand nach der Nummer als Bestandteil der Nummer setzen	19	<code>\scrjura@special@reset@par:</code> neue Anweisung (intern) . .	24
Zähler <code>contractAbsoluteParagraph</code> wird erhöht	18	<code>\thisparnumber:</code> neue Anweisung	24
<code>\contract@subparagraph:</code> Abstand nach der Nummer als Bestandteil der Nummer setzen	20	<code>\toclevel@cpar:</code> neue für hyperref	12
Zähler <code>contractAbsoluteParagraph</code> wird erhöht	20	<code>juratocindent:</code> Verwendung der renovierten Schnittstelle mit <code>\FamilyKeyStateProcessed</code>	12
<code>ref:</code> Verwendung der renovierten Schnittstelle mit <code>\FamilyKeyStateProcessed</code>	14	<code>juratoclevel:</code> Verwendung der renovierten Schnittstelle mit <code>\FamilyKeyState</code>	11
<code>\ellipsispar:</code> neu	15	<code>juratocnumberwidth:</code> Verwendung der renovierten Schnittstelle mit <code>\FamilyKeyStateProcessed</code>	12
<code>contractAbsoluteParagraph:</code> neuer (interner) Zähler über alle Paragraphen, um ggf. die Zählung neu beginnen zu können	16	<code>v0.7a</code>	
<code>\parellipsis:</code> neu	15	<code>\contract@sentence:</code> <code>\nobreak\hskip\z@</code> eingefügt, um die Trennung des ersten Worts nach der Satzmarkierung zu ermöglichen	30
		auf <code>\textsuperscript</code> umgestellt	30