

KOMA-Script Datei `scrjura.dtx` *

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

2010/06/07

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 Kolumnentitel	5
2.4	Der Paragraph	5
2.5	Verweise	6
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 v0.5d 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` 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 erkaufte 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`, `sentencesshort`, `sentencesnumeric`, `nosentence` möglich. Beispiel: `\usepackage[ref=parlong,ref=sentenceshort]{scrjura}`. Wir haben allerdings diese Kombinationen nicht alle getestet.

2.3 Optionen betreffend Kolumnentitel

Bei langen Verträgen können automatische Kolumnentitel hilfreich sein. Allerdings hängt die Ausgabe hier davon ab, wie die diese erzeugt werden. `paragraphmark` unterstützt lebende Kolumnentitel. 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>	Kolumnentitel 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.

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 Absatz 1 Satz 1
<code>\refL{testA}</code>	§ 1 Absatz 1 Satz 1
<code>\refS{testA}</code>	§ 1 Abs. 1 S. 1
<code>\refN{testA}</code>	§ 1 I 1.
<code>\refParagraph{testA}</code>	§ 1
<code>\refParagraphN{testA}</code>	1
<code>\refPar{testA}</code>	Absatz 1
<code>\refParL{testA}</code>	Absatz 1
<code>\refParS{testA}</code>	Abs. 1
<code>\refParN{testA}</code>	I
<code>\refParN[arabic]{testA}</code>	1
<code>\refSentence{testA}</code>	Satz 1
<code>\refSentenceL{testA}</code>	Satz 1
<code>\refSentenceS{testA}</code>	S. 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 Satz 1
<code>\refPar{testA}</code>	
<code>\refSentence{testA}</code>	Satz 1

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

Code	Ergebnis
<code>\ref{testA}</code>	§ 1 Absatz 1
<code>\refPar{testA}</code>	Absatz 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>	
<code>\Par</code>	Absatznummern ein- und ausschalten.
<code>\ifparnumber</code>	Zähler für die Absatznummern und dessen Ausgabe und Formatierung und
<code>\parnumbertrue</code>	Formatierung der Referenz. ACHTUNG: <code>\thepar</code> muss bei unverändertem
<code>\parnumberfalse</code>	<code>\parciteformat</code> eine arabische Zahl sein!
<code>par</code>	
<code>\thepar</code>	So etwas wie <code>\ref</code> nur für den Paragraphen.
<code>\parformat</code>	So etwas wie <code>\ref</code> nur für den Absatz.
<code>\parciteformat</code>	So etwas wie <code>\ref</code> nur für den Satz.
<code>\refParagraph</code>	
<code>\refPar</code>	Zähler für die Satznummer und dessen Ausgabe, sowie Formatierung der
<code>\refSentence</code>	Referenz. ACHTUNG: <code>\thesentence</code> muss bei unverändertem <code>\sentenceciteformat</code>
<code>sentence</code>	eine arabische Zahl sein!
<code>\thesentence</code>	
<code>\sentenceciteformat</code>	Werden intern verwendet und müssen deshalb allenfalls für Paketautoren dokumentiert werden.
<code>\newmaxpar</code>	
<code>\getmaxpar</code>	

3 Bekannte Probleme

Die Verweisteknik 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}
```

`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.

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

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

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

```
14 \KOMA@ifkey{juratotoc}{@juratotoc}
```

`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.

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

paragraphmark Mit diesen Optionen wird \Paragraphmark so umdefiniert, dass es ent-
markright weder 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!)

```

16 \newcommand*{\Paragraphmark}[1]{%
17 \KOMA@key{paragraphmark}{%
18   \begingroup
19     \KOMA@set@ncmdkey{paragraphmark}{@tempa}{%
20       {false}{0},{off}{0},{no}{0},%
21       {forceright}{1},%
22       {forceboth}{2},%
23       {right}{3},%
24       {both}{4}%
25     }{#1}%
26   \ifcase\number\@tempa
27     \endgroup
28     \let\Paragraphmark\@gobble
29   \or
30     \endgroup
31     \renewcommand*{\Paragraphmark}[1]{\markright{##1}}%
32   \or
33     \endgroup
34     \renewcommand*{\Paragraphmark}[1]{\markboth{##1}{##1}}%
35   \or
36     \endgroup
37     \renewcommand*{\Paragraphmark}[1]{%
38       \ifx\@mkboth\@gobbletwo\else\markright{##1}\fi}%
39   \or
40     \endgroup
41     \renewcommand*{\Paragraphmark}[1]{\@mkboth{##1}{##1}}%
42   \else
43     \endgroup
44   \fi
45 }
46 \KOMA@DeclareDeprecatedOption[scrjura]{markright}{paragraphmark=forceright}
47 \KOMA@DeclareDeprecatedOption[scrjura]{markboth}{paragraphmark=forceboth}

```

ref Die Form, in der Absätze und Sätze referenziert werden. Es gibt eine lange,
parcitenam eine abgekürzte und eine numerische Form.
sentencecitename

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

```

48 \newcommand*{\parcite@format}{0}
49 \newcommand*{\sentencecite@format}{0}

```

Per Option kann die Voreinstellung verändert werden.

```

50 \KOMA@key{ref}{%
51   \begingroup
52     \KOMA@set@ncmdkey{ref}{@tempa}{%
53       {parlong}{1},{longpar}{1},{ParL}{1},%
54       {parshort}{2},{shortpar}{2},{ParS}{2},%
55       {parnumeric}{3},{numericpar}{3},{ParN}{3},%
56       {paroff}{4},{nopar}{4},%
57       {sentencelong}{10},{longsentence}{10},{SentenceL}{10},%
58       {sentenceshort}{20},{shortsentence}{20},{SentenceS}{20},%
59       {sentencenumeric}{30},{numericssentence}{30},{SentenceN}{30},%
60       {sentenceoff}{40},{nosentence}{40},%
61       {long}{11},%
62       {short}{22},%
63       {numeric}{33},%
64       {paragraphonly}{44},{ParagraphOnly}{44},{OnlyParagraph}{44}%
65     }{#1}%
66     \@tempcnta=\@tempa\relax
67     \@tempcntb=\z@
68     \@whilenum \@tempcnta>9 \do{%
69       \advance\@tempcnta -10\relax
70       \advance\@tempcntb \@ne\relax
71     }%
72     \ifcase \@tempcnta
73     \or
74       \aftergroup\def\aftergroup\parcite@format
75       \aftergroup{\aftergroup0\aftergroup}%
76     \or
77       \aftergroup\def\aftergroup\parcite@format
78       \aftergroup{\aftergroup1\aftergroup}%
79     \or
80       \aftergroup\def\aftergroup\parcite@format
81       \aftergroup{\aftergroup2\aftergroup}%
82     \or
83       \aftergroup\def\aftergroup\parcite@format
84       \aftergroup{\aftergroup-\aftergroup1\aftergroup}%
85     \fi
86     \ifcase \@tempcntb
87     \or
88       \aftergroup\def\aftergroup\sentencecite@format
89       \aftergroup{\aftergroup0\aftergroup}%
90     \or
91       \aftergroup\def\aftergroup\sentencecite@format
92       \aftergroup{\aftergroup1\aftergroup}%
93     \or
94       \aftergroup\def\aftergroup\sentencecite@format
95       \aftergroup{\aftergroup2\aftergroup}%
96     \or
97       \aftergroup\def\aftergroup\sentencecite@format

```

```

98      \aftergroup{\aftergroup-\aftergroup1\aftergroup}%
99      \fi
100     \endgroup
101  }
102  \KOMACodeclareDeprecatedOption[scrjura]{parcitename}{ref=parlong}
103  \KOMACodeclareDeprecatedOption[scrjura]{sentencecitename}{ref=sentencelong}

```

Optionen ausführen.

```

104  \KOMAProcessOptions\relax

```

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

```

105  \newcommand*{\jura@env@type}{}

```

`contract` Die Umgebung `contract` wird als Anweisung `\contract` 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.

`\contract`

```

106  \newcommand*{\contract}{%
107    \ifx\jura@env@type\@empty
108      \let\@doendpe\contract@doendpe
109      \let\Paragraph\contract@paragraph
110      \let\c@Paragraph\c@contractParagraph
111      \edef\cl@Paragraph{\cl@Paragraph\cl@contractParagraph}%
112      \let\SubParagraph\contract@subparagraph
113      \let\c@SubParagraph\c@contractSubParagraph
114      \edef\cl@SubParagraph{\cl@SubParagraph\cl@contractSubParagraph}%
115      \let\Par\contract@everypar
116      \let\Sentence\contract@sentence
117      \renewcommand*{\jura@env@type}{contract}%
118    \else
119      \PackageError{contract}{nested 'contract' detected}{%
120        You may not use a 'contract' environment inside\MessageBreak
121        a '\jura@env@type' environment or after loading\MessageBreak
122        package 'scrjura' with option '\jura@env@type'!}%
123    \fi
124  }
125  \let\if@jura@skiphyperref\iftrue
126  \let\cl@Paragraph\@empty
127  \let\cl@SubParagraph\@empty
128  \newcounter{contractParagraph}
129  \renewcommand*{\thecontractParagraph}{%
130    {\contract@Paragraphformat{\arabic{contractParagraph}}}}

```

```

131 \DeclareRobustCommand*\contract@Paragraphformat}[1]{\paragraphformat{#1}}
132 \newcommand*\paragraphformat}[1]{\S~#1}
133 \newcounter{contractSubParagraph}
134 \@addtoreset{SubParagraph}{Paragraph}
135 \renewcommand*\thecontractSubParagraph{%
136   {\theParagraph\alph{contractSubParagraph}}}

```

`\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 beiden aber auf leer setzen. Insbesondere aber kann man sie mit den `nohead` Optionen auch abschalten.
`noentry`

```

137 \define@key{contract}{title}{%
138   \def\contract@title{#1}%
139   \ifx\contract@entry\relax\def\contract@entry{\contract@title}\fi
140   \ifx\contract@head\relax\def\contract@head{\contract@title}\fi
141 }
142 \define@key{contract}{entry}{\def\contract@entry{#1}}
143 \define@key{contract}{noentry}[]{\let\contract@entry\relax}
144 \define@key{contract}{head}{\def\contract@head{#1}}
145 \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.

```

146 \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`

```

147 \newcommand*\contract@preskip}{2\baselineskip}
148 \newcommand*\contract@postskip}{\baselineskip}
149 \define@key{contract}{preskip}{\def\contract@preskip{#1}}
150 \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.

```
151 \newif\ifcontract@dummy
152 \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!

```
153 \newcommand*{\contract@paragraph@font}{\sffamily\bfseries\large\@hangfrom}
```

```

@AbsParagraph
\theH@AbsParagraph
\theHParagraph
\theHSubParagraph
154 % \textsf{hyperref}-Code, der noch nicht getestet ist!
155 \newcounter{@AbsParagraph}
156 \newcommand*{\theH@AbsParagraph}{P-\arabic{@AbsParagraph}}
157 \newcommand*{\theHParagraph}{\theH@AbsParagraph}
158 \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.

```
159 \newcommand*{\contract@paragraph}[1]{%
160   \ManualPar\parnumbertrue
161   \let\contract@title\relax
162   \let\contract@entry\relax
163   \let\contract@head\relax
164   \let\contract@number\relax
165   \contract@dummyfalse
166   \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.

```
167 \ifcontract@dummy\else
168   \@afterindentfalse
169   \addvspace{\contract@preskip}%
170 \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.


```

171 \ifx\contract@number\relax
172   \let\p@Paragraph\@empty
173   \let\theParagraph\thecontractParagraph
174   \refstepcounter{Paragraph}%
175 \else
176   \begingroup
177   \let\@elt\@stpelt
178   \cl@Paragraph
179 \endgroup
180 \protected@edef\theParagraph{\S~\contract@number}%
181 \protected@edef\@currentlabel{\theParagraph}%
182 \fi
183 \stepcounter{@AbsParagraph}%
184 \begingroup\expandafter\expandafter\expandafter\endgroup
185 \expandafter\ifx\csname if@skiphyperref\endcsname\relax
186 \else
187   \expandafter\let\csname if@jura@skiphyperref\endcsname
188   \csname if@skiphyperref\endcsname
189 \fi
190 \if@jura@skiphyperref\else
191   \hyper@refstepcounter{@AbsParagraph}%
192   \typeout{absolute Nummer: \the@AbsParagraph^^JLabel: '\@currentHref'}%
193 \fi

```

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

```

194 \let\theSubParagraph\theParagraph

```

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

```

195 \ifcontract@dummy\else
196   \begingroup
197   \if@juratitlepagebreak\else\interlinepenalty\@M\fi
198   \contract@paragraph@font \theParagraph
199   \ifx\contract@title\relax\else\enskip\contract@title\fi
200   \ifx\contract@entry\relax\else
201     \addcontentsline{toc}{cpar}{%
202       \protect\numberline{\theParagraph}\contract@entry
203     }%
204     \addcontentsline{cpar}{cpar}{%
205       \protect\numberline{\theParagraph}\contract@entry
206     }%
207   \fi
208   \ifx\contract@head\relax\else
209     \expandafter\Paragraphmark\expandafter{%
210       \expandafter\theSubParagraph\expandafter\enskip\contract@head}%
211   \fi
212   \par
213 \endgroup\nobreak\vskip\contract@postskip

```

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

```

214 \jura@afterheading{contract}%
215 \fi
216 }

```

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

```

217 \newcommand*{\contract@subparagraph}[1]{%
218 \ManualPar\parnumbertrue
219 \let\contract@title\relax
220 \let\contract@entry\relax
221 \let\contract@head\relax
222 \let\contract@number\relax
223 \contract@dummyfalse
224 \ifx\relax#1\relax\else\setkeys{contract}{#1}\fi
225 \ifcontract@dummy\else
226 \afterindentfalse
227 \vskip\contract@preskip
228 \fi
229 \ifx\contract@number\relax
230 \let\p@SubParagraph\empty
231 \let\theSubParagraph\thecontractSubParagraph
232 \refstepcounter{SubParagraph}%
233 \else
234 \begingroup
235 \let\@elt\@stpelt
236 \cl@SubParagraph
237 \endgroup
238 \protected@edef\theSubParagraph{\theParagraph\contract@number}%
239 \protected@edef\@currentlabel{\theSubParagraph}%
240 \fi
241 \stepcounter{@AbsParagraph}%
242 \begingroup\expandafter\expandafter\expandafter\endgroup
243 \expandafter\ifx\csname if@skiphyperref\endcsname\relax
244 \else
245 \expandafter\let\csname if@jura@skiphyperref\endcsname
246 \csname if@skiphyperref\endcsname
247 \fi
248 \if@jura@skiphyperref\else
249 \hyper@refstepcounter{@AbsParagraph}%
250 \typeout{absolute Nummer: \the@AbsParagraph^^JLabel: '\@currentHref'}%
251 \fi
252 \ifcontract@dummy\else
253 \begingroup
254 \if@juratitlepagebreak\else\interlinepenalty\@M\fi
255 \contract@paragraph@font \theSubParagraph

```

```

256     \ifx\contract@title\relax\else\enskip\contract@title\fi
257     \ifx\contract@entry\relax\else
258         \addcontentsline{toc}{cpar}{%
259             \protect\numberline{\theSubParagraph}\contract@entry
260         }%
261         \addcontentsline{cpa}{cpar}{%
262             \protect\numberline{\theSubParagraph}\contract@entry
263         }%
264     \fi
265     \ifx\contract@head\relax\else
266         \expandafter\Paragraphmark\expandafter{%
267             \expandafter\theSubParagraph\expandafter\enskip\contract@head}%
268     \fi
269     \par
270 \endgroup
271 \nobreak\vskip\contract@postskip
272 \jura@afterheading{contract}%
273 \fi
274 }

```

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

```

275 \newcommand*{\AutoPar}{%
276     \expandafter\let\expandafter\jura@everypar
277     \csname \jura@type @everypar\endcsname
278 }
279 \newcommand*{\ManualPar}{%
280     \let\jura@everypar\relax
281 }

```

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

```

282 \newcommand*{\jura@afterheading}[1]{%
283     \@nobreaktrue
284     \everypar{%
285         \if@nobreak
286             \@nobreakfalse
287             \clubpenalty \@M
288             \if@afterindent \else
289                 {\setbox\z@\lastbox}%
290             \fi
291         \else
292             \clubpenalty \@clubpenalty
293             \everypar{%
294                 \jura@everypar
295             }%

```

```

296     \fi
297     \jura@everypar
298 }%
299 \def\jura@type{#1}%
300 \AutoPar
301 }

```

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

```

302 \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.

```

303 \CheckCommand*\@doendpe{\@endpetrue
304   \def\par{\@restorepar\everypar}\par\@endpefalse}\everypar
305   {\setbox\z@\lastbox}\everypar}\@endpefalse}}
306 \newcommand*\contract@doendpe{%
307   \@endpetrue
308   \def\par{%
309     \@restorepar\everypar{%
310       \csname jura@everypar\endcsname
311     }%
312     \par\@endpefalse
313   }%
314   \everypar{%
315     {\setbox\z@\lastbox}\everypar{%
316       \csname jura@everypar\endcsname
317     }%
318     \@endpefalse
319   }%
320 }

```

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

```

321 \newcommand*\l@cpa[2]{%
322   \ifjuratotoc
323     \dottedtocline{\value{juratoclevel}}{1.5em}{2em}{#1}{#2}%
324   \fi
325 }

```

`juratoclevel` Dieser Zähler gibt an, in welcher Ebene die Verzeichniseinträge für die Paragraphen erfolgen. Voreinstellung ist Ebene 2.

```

326 \newcounter{juratoclevel}
327 \setcounter{juratoclevel}{2}%

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

328 \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.

329 \DeclareRobustCommand*\jura@usetype}[1]{\jura@@usetype{#1}}
330 \newcommand*\jura@@usetype}[1]{

\contract@everypar \jura@everpar für Verträge.

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

331 \newif\ifparnumber
332 \newcounter{par}
333 \renewcommand*\{thepar}\{arabic{par}}
334 \newcommand*\{theHpar}\{theH@AbsParagraph-\Roman{par}}
335 \newcommand*\{parformat}\{(\thepar)}
336 \renewcommand*\p@par{\jura@usetype{\jura@env@type}\theSubParagraph\jura@separator}
337 \@addtoreset{par}{Paragraph}
338 \@addtoreset{par}{SubParagraph}

339 \newcommand*\{contract@everypar}\{
340   \ifparnumber
341     \refstepcounter{par}%
342     \refstepcounter{sentence}%
343     \begingroup
344       \protected@write\@auxout{\let\contract@Paragraphformat\@firstofone}\{
345         \string\newmaxpar{contract}\{theSubParagraph}\{arabic{par}}\}%
346       \getmaxpar\@tempa{contract}\{theSubParagraph}%
347       \typeout{Stored max is \@tempa}%

```

```

348     \ifnum \@tempa>\@ne
349     \parformat\nobreakspace
350     \fi
351 \endgroup
352 \else
353     \setcounter{sentence}{0}\refstepcounter{sentence}%
354 \fi
355 }

```

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

```

\ref@L
356 \newcommand*{\refL}{\@ifstar {\ref@L*}{\ref@L{}}}
357 \newcommand*{\ref@L}[2]{%
358     \begingroup
359     \def\parcite@format{0}%
360     \let\sentencecite@format\parcite@format
361     \ref#1{#2}%
362 \endgroup
363 }

```

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

```

\ref@S
364 \newcommand*{\refS}{\@ifstar {\ref@S*}{\ref@S{}}}
365 \newcommand*{\ref@S}[2]{%
366     \begingroup
367     \def\parcite@format{1}%
368     \let\sentencecite@format\parcite@format
369     \ref#1{#2}%
370 \endgroup
371 }

```

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

```

\ref@N
372 \newcommand*{\refN}{\@ifstar {\ref@N*}{\ref@N{}}}
373 \newcommand*{\ref@N}[2]{%
374     \begingroup
375     \def\parcite@format{2}%
376     \let\sentencecite@format\parcite@format
377     \ref#1{#2}%
378 \endgroup
379 }

```

`\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.

```

380 \newcommand*{\refParagraph}{%
381     \@ifstar {\ref@Paragraph*}{\ref@Paragraph{}}

```

```

382 }
383 \newcommand*{\ref@Paragraph}[2]{%
384   \expandafter\ifx\csname r@#2\endcsname\relax
385     \ref#1{#2}%
386   \else
387     \begingroup

```

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

```

388     \expandafter\expandafter\expandafter\expandafter
389     \expandafter\expandafter\expandafter\def
390     \expandafter\expandafter\expandafter\expandafter
391     \expandafter\expandafter\expandafter\@tempb
392     \expandafter\expandafter\expandafter\expandafter
393     \expandafter\expandafter\expandafter{%
394     \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%

```

Den ersten Teil der Referenz in \@tempa ablegen.

```

395     \def\@tempc##1##2\@nil{##1}%
396     \let\jura@separator\@gobble
397     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
398     \csname r@#2\endcsname\noexpand\@nil}%

```

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

```

399     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
400     \@tempa\@nil}%
401     \let\@@protect\protect
402     \let\protect\noexpand
403     \expandafter\edef\csname r@#2\endcsname{\@tempa\@tempb}%
404     \let\protect\@@protect
405     \ref#1{#2}%
406   \endgroup
407 \fi
408 }

```

`\refParagraphN` 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.

```

409 \newcommand*{\refParagraphN}{%
410   \@ifstar {\ref@ParagraphN*}{\ref@ParagraphN{}}
411 }
412 \newcommand*{\ref@ParagraphN}[2]{%
413   \begingroup
414     \let\paragraphformat\relax
415     \ref@Paragraph{#1}{#2}%
416   \endgroup
417 }

```

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

```

418 \newcommand*{\refPar}{%
419   \@ifstar {\ref@Par*}{\ref@Par{}}
420 }
421 \newcommand*{\ref@Par}[2]{%
422   \expandafter\ifx\csname r@#2\endcsname\relax
423     \ref#1{#2}%
424   \else
425     \begingroup

```

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

```

426     \expandafter\expandafter\expandafter\expandafter
427     \expandafter\expandafter\expandafter\def
428     \expandafter\expandafter\expandafter\expandafter
429     \expandafter\expandafter\expandafter\@tempb
430     \expandafter\expandafter\expandafter\expandafter
431     \expandafter\expandafter\expandafter{%
432     \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%

```

Den ersten Teil der Referenz in `\@tempa` ablegen.

```

433     \def\@tempc##1##2\@nil{##1}%
434     \let\jura@separator\@gobble
435     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
436       \csname r@#2\endcsname\noexpand\@nil}%

```

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

```

437     \def\@tempc##1##2##3\@nil{##2}%
438     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
439       \@tempa{%
440         \protect\G@refundefinedtrue
441         \nfss@text{\reset@font\bfseries ??}%
442         \@latex@warning{Reference ‘#2’ on page \thepage \space
443           with undefined par number}%
444       }\noexpand\@nil}%
445     \let\@@protect\protect
446     \let\protect\noexpand
447     \expandafter\edef\csname r@#2\endcsname{{\@tempa}\@tempb}%
448     \let\protect\@@protect
449     \ref#1{#2}%
450   \endgroup
451 \fi
452 }

```



```

\refParL
\ref@ParX
453 % Das Gleiche zwingend lang.
454 \newcommand*{\refParL}{%
455   \@ifstar {\ref@ParX0*}{\ref@ParX0{}}
456 }
457 \newcommand*{\ref@ParX}[3]{%
458   \begingroup
459     \def\parcite@format{#1}%
460     \let\sentencecite@format\parcite@format
461     \ref@Par{#2}{#3}%
462   \endgroup
463 }

```

\refParS Das Gleiche zwingend kurz.

```

464 \newcommand*{\refParS}{%
465   \@ifstar {\ref@ParX1*}{\ref@ParX1{}}
466 }

```

\refParN Das Gleiche zwingend numerisch.

```

467 \newcommand*{\refParN}{%
468   \@ifstar {\ref@ParN2*}{\ref@ParN2{}}
469 }
470 \newcommand*{\ref@ParN}[2]{%
471   \@ifnextchar [%]
472     {\ref@ParN{#1}{#2}}%
473     {\ref@ParX{#1}{#2}}%
474 }
475 \newcommand*{\ref@ParN}{%
476   \def\ref@ParN#1#2[#3]#4{%
477     \begingroup
478       \renewcommand*{\parnumericformat}[1]{%
479         \csname @#3\endcsname{\number ##1\relax}%
480       }%
481       \ref@ParX{#1}{#2}{#4}%
482     \endgroup
483 }

```

\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.

```

484 \newcommand*{\refSentence}{%
485   \@ifstar {\ref@Sentence*}{\ref@Sentence{}}
486 }
487 \newcommand*{\ref@Sentence}[2]{%

```

```

488 \expandafter\ifx\csname r@#2\endcsname\relax
489   \ref#1{#2}%
490 \else
491   \begingroup

```

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

```

492   \expandafter\expandafter\expandafter\expandafter
493   \expandafter\expandafter\expandafter\def
494   \expandafter\expandafter\expandafter\expandafter
495   \expandafter\expandafter\expandafter\@tempb
496   \expandafter\expandafter\expandafter\expandafter
497   \expandafter\expandafter\expandafter{%
498   \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%

```

Den ersten Teil der Referenz in \@tempa ablegen.

```

499   \def\@tempc##1##2\@nil{##1}%
500   \let\jura@separator\@gobble
501   \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
502   \csname r@#2\endcsname\noexpand\@nil}%

```

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

```

503   \def\@tempc##1##2##3##4\@nil{##3}%
504   \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
505   \@tempa}{%
506   \protect\G@refundefinedtrue
507   \nfss@text{\reset@font\bfseries ??}%
508   \@latex@warning{Reference ‘#2’ on page \thepage \space
509   with undefined sentence number}%
510   }\noexpand\@nil}%
511   \let\@@protect\protect
512   \let\protect\noexpand
513   \expandafter\edef\csname r@#2\endcsname{{\@tempa}\@tempb}%
514   \let\protect\@@protect
515   \ref#1{#2}%
516 \endgroup
517 \fi
518 }

```

```

\refSentenceL
\ref@SentenceX

```

```

519 % Das Gleiche zwingend lang.
520 \newcommand*{\refSentenceL}{%
521   \@ifstar {\ref@SentenceX0*}{\ref@SentenceX0{}}
522 }
523 \newcommand*{\ref@SentenceX}[3]{%
524   \begingroup
525   \def\parcite@format{#1}%

```

```

526 \let\sentencecite@format\parcite@format
527 \ref@Sentence{#2}{#3}%
528 \endgroup
529 }

```

`\refSentenceS` Das Gleiche zwingend kurz.

```

530 \newcommand*{\refSentenceS}{%
531 \ifstar {\ref@SentenceX1*}{\ref@SentenceX1{}}
532 }

```

`\refSentenceN` Das Gleiche zwingend numerisch.

```

533 \newcommand*{\refSentenceN}{%
534 \ifstar {\ref@SentenceX2*}{\ref@SentenceX2{}}
535 }

```

`\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
536 \newcounter{sentence}[par]
537 \renewcommand*{\thesentence}{\arabic{sentence}}
538 \newcommand*{\theHsentence}{\theHpar-\arabic{sentence}}
539 \renewcommand*{\p@sentence}{\expandafter\p@@sentence}
540 \newcommand*{\p@@sentence}[1]{\p@par{\par@cite{\thepar}}%
541 \jura@separator{\nobreakspace}}{\sentence@cite{#1}}}
542 \newcommand*{\contract@sentence}{%
543 \refstepcounter{sentence}%
544 \ensuremath{\sim{\mbox{\fontsize\sfontsize\selectfont\thesentence}}}%
545 }

```

`\parciteformat` Zitierstil für Absätze.

```

546 \DeclareRobustCommand*{\par@cite}[1]{\parciteformat{#1}}
547 \newcommand*{\parciteformat}[1]{%
548 \ifcase \parcite@format
549 \expandafter\parlongformat
550 \or
551 \expandafter\parshortformat
552 \or
553 \expandafter\parnumericformat
554 \else
555 \unskip\expandafter\@gobble
556 \fi
557 {#1}%

```

558 }

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

```
559 \DeclareRobustCommand*\sentence@cite}[1]{\sentenceciteformat{#1}}
560 \newcommand*\sentenceciteformat[1]{%
561   \ifcase \sentencecite@format
562     \expandafter\sentencelongformat
563   \or
564     \expandafter\sentenceshortformat
565   \or
566     \expandafter\sentencenumericformat
567   \else
568     \unskip\expandafter\@gobble
569   \fi
570   {#1}%
571 }
```

`\parlongformat` Formatierung in den sechs Formen.

```
\parshortformat
\parnumericformat 572 \newcommand*\parlongformat[1]{\parname~#1}
573 \newcommand*\parshortformat[1]{\parshortname~#1}
\sentencelongformat 574 \newcommand*\parnumericformat[1]{\@Roman{\number #1\relax}}
\sentenceshortformat 575 \newcommand*\sentencelongformat[1]{\sentencename~#1}
\sentencenumericformat 576 \newcommand*\sentenceshortformat[1]{\sentenceshortname~#1}
577 \newcommand*\sentencenumericformat[1]{\@arabic{\number #1\relax}.}
```

`\parname` Der Name eines Absatzes und eines Satzes.

```
\parshortname
\sentencename 578 \newcommand*\parname{\Paragraph}
579 \AtBeginDocument{%
\sentenceshortname 580   \providecaptionname{german}\parname{Absatz}%
\scrjura@lang@error 581   \providecaptionname{german}\parshortname{Abs.}%
582   \providecaptionname{german}\sentencename{Satz}%
583   \providecaptionname{german}\sentenceshortname{S.}%
584   \providecaptionname{ngerman}\parname{Absatz}%
585   \providecaptionname{ngerman}\parshortname{Abs.}%
586   \providecaptionname{ngerman}\sentencename{Satz}%
587   \providecaptionname{ngerman}\sentenceshortname{S.}%
588 }
589 \providecommand*\parname{\scrjura@lang@error{\parname}}
590 \providecommand*\parshortname{\scrjura@lang@error{\parshortname}}
591 \providecommand*\sentencename{\scrjura@lang@error{\sentencename}}
592 \providecommand*\sentenceshortname{\scrjura@lang@error{\sentenceshortname}}
593 \newcommand*\scrjura@lang@error[1]{%
594   \PackageErrorNoLine{scrjura}{%
595     current language not supported%
596   }{%
```

```

597     Currently scrjura only supports languages ‘‘german’’ and
598     ‘‘ngerman’’. \MessageBreak
599     It seems, that you are using another language (maybe
600     \language). \MessageBreak
601     Because of this you have to define \string#1\space by
602     yourself! \MessageBreak
603     It would be nice if you’ll send your definitions to the author.%
604 }%
605 \textbf{??}%
606 }

```

`\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.

```

607 \newcommand*{\newmaxpar}[3]{%
608   \begingroup
609     \protected@edef\@tempa{#2}\@onelevel@sanitize\@tempa
610     \expandafter\xdef\csname max@#1\@tempa\endcsname{#3}%
611   \endgroup
612 }
613 \newcommand*{\getmaxpar}[3]{%
614   \begingroup
615     \expandafter\let\csname #2@Paragraphformat\endcsname\@firstofone
616     \edef\@tempa{#3}%
617     \@onelevel@sanitize\@tempa
618     \expandafter\ifx \csname max@#2\@tempa\endcsname\relax
619       \edef\@tempa{\endgroup\edef\noexpand#1{\expandafter\the\value{par}}}%
620     \else
621       \edef\@tempa{\endgroup
622         \edef\noexpand#1{\csname max@#2\@tempa\endcsname}}%
623     \fi
624   \@tempa
625 }

```

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		C	
<code>@AbsParagraph</code> (Zähler)	<u>154</u>	<code>\contract</code>	<u>106</u>
<code>\@doendpe</code>	<u>303</u>	<code>contract</code> (Umgebung)	<u>106</u>
A		<code>contract</code> (Option)	<u>0</u> , <u>11</u>
		<code>\contract@everypar</code>	<u>331</u>
<code>\AutoPar</code>	<i>10</i> , <u>275</u>	<code>\contract@paragraph</code>	<u>137</u>

<code>\contract@paragraph@font</code> . . .	<u>153</u>
<code>\contract@postskip</code>	<u>147</u>
<code>\contract@preskip</code>	<u>147</u>
<code>\contract@sentence</code>	<u>536</u>
<code>\contract@subparagraph</code>	<u>217</u>

D

<code>dummy</code> (Option)	<u>151</u>
---------------------------------------	------------

E

<code>entry</code> (Option)	<u>137</u>
---------------------------------------	------------

G

<code>\getmaxpar</code>	<i>10</i> , <u>607</u>
-----------------------------------	------------------------

H

<code>head</code> (Option)	<u>137</u>
--------------------------------------	------------

I

<code>\if@juratitlepagebreak</code>	<u>15</u>
<code>\if@juratotoc</code>	<u>14</u>
<code>\ifcontract@dummy</code>	<u>151</u>
<code>\ifparnumber</code>	<i>10</i> , <u>331</u>

J

<code>\jura@@usetype</code>	<u>329</u>
<code>\jura@afterheading</code>	<u>282</u>
<code>\jura@env@type</code>	<u>105</u>
<code>\jura@everypar</code>	<u>302</u>
<code>\jura@separator</code>	<u>328</u>
<code>\jura@usetype</code>	<u>329</u>
<code>juratitlepagebreak</code> (Option) .	<u>0</u> , <u>15</u>
<code>\juratoclevel</code>	<u>4</u>
<code>juratoclevel</code> (Zähler)	<u>326</u>
<code>juratotoc</code> (Option)	<u>0</u> , <u>14</u>

L

<code>\l@cpar</code>	<u>321</u>
--------------------------------	------------

M

<code>\ManualPar</code>	<i>10</i> , <u>275</u>
<code>markboth</code> (Option)	<u>16</u>
<code>markright</code> (Option)	<u>16</u>

N

<code>\newmaxpar</code>	<i>10</i> , <u>607</u>
<code>noentry</code> (Option)	<u>137</u>
<code>nohead</code> (Option)	<u>137</u>
<code>number</code> (Option)	<u>146</u>

O

Optionen:

<code>contract</code>	<u>0</u> , <u>11</u>
<code>dummy</code>	<u>151</u>
<code>entry</code>	<u>137</u>
<code>head</code>	<u>137</u>
<code>juratitlepagebreak</code>	<u>0</u> , <u>15</u>
<code>juratotoc</code>	<u>0</u> , <u>14</u>
<code>markboth</code>	<u>16</u>
<code>markright</code>	<u>16</u>
<code>noentry</code>	<u>137</u>
<code>nohead</code>	<u>137</u>
<code>number</code>	<u>146</u>
<code>paragraphmark</code>	<u>0</u> , <u>16</u>
<code>parcitename</code>	<u>48</u>
<code>postskip</code>	<u>0</u> , <u>147</u>
<code>preskip</code>	<u>0</u> , <u>147</u>
<code>ref</code>	<u>48</u>
<code>sentencecitename</code>	<u>48</u>
<code>title</code>	<u>137</u>

P

<code>\p@par</code>	<u>331</u>
<code>\p@sentence</code>	<u>536</u>
<code>\Par</code>	<i>10</i>
<code>\par</code>	<i>10</i>
<code>par</code> (Zähler)	<u>331</u>
<code>\Paragraph</code>	<u>5</u>
<code>\Paragraphmark</code>	<u>16</u>
<code>paragraphmark</code> (Option) . . .	<u>0</u> , <u>16</u>
<code>\parcite@fromat</code>	<u>48</u>
<code>\parciteformat</code>	<i>10</i> , <u>546</u>
<code>parcitename</code> (Option)	<u>48</u>
<code>\parformat</code>	<i>10</i>
<code>\parlongformat</code>	<u>572</u>
<code>\parname</code>	<i>10</i> , <u>578</u>
<code>\parnumberfalse</code>	<i>9</i> , <i>10</i>
<code>\parnumbertrue</code>	<i>9</i> , <i>10</i>
<code>\parnumericformat</code>	<u>572</u>
<code>\parshortformat</code>	<u>572</u>
<code>\parshortname</code>	<u>578</u>
<code>postskip</code> (Option)	<u>0</u> , <u>147</u>
<code>preskip</code> (Option)	<u>0</u> , <u>147</u>

R

<code>ref</code> (Option)	<u>48</u>
<code>\ref@L</code>	<u>356</u>
<code>\ref@N</code>	<u>372</u>
<code>\ref@Par</code>	<u>418</u>
<code>\ref@Paragraph</code>	<u>380</u>

\ref@ParagraphN	409	\sentencename	10, 578
\ref@ParX	453	\sentencenumericformat	572
\ref@S	364	\sentenceshortformat	572
\ref@Sentence	484	\sentenceshortname	578
\ref@SentenceX	519	\SubParagraph	6
\refL	356		
\refN	372	T	
\refPar	10, 418	\thecontractParagraph	10
\refParagraph	10, 380	\thecontractSubParagraph	10
\refParagraphN	409	\theH@AbsParagraph	154
\refParL	453	\theHpar	331
\refParN	467	\theHParagraph	154
\refParS	464	\theHSentence	536
\refS	364	\theHSubParagraph	154
\refSentence	10, 484	\thepar	10, 331
\refSentenceL	519	\thesentence	10, 536
\refSentenceN	533	title (Option)	137
\refSentencesS	530		
S		U	
\scrjura@lang@error	578	Umgebungen:	
\Sentence	9	contract	106
\sentence	10	Z	
sentence (Zähler)	536	Zähler:	
\sentencecite@fromat	48	@AbsParagraph	154
\sentenceciteformat	10, 559	juratoclevel	326
sentencecitename (Option)	48	par	331
\sentencelongformat	572	sentence	536

0.5c		ment	29
	<code>\contract@everypar: \contract@Paragraph</code>	<code>\footnoted@edef</code>	durch
	expandiert beim Schreiben zu	<code>\edef</code> ersetzt	29
	seinem Argument	v0.5d	
	erstes Argument von	<code>\contract: \jura@env@type</code>	
	<code>\newmaxpar</code> und zweites Ar-	wird gesetzt	14
	gument von <code>\getmaxpar</code>	fehlende Fehlermeldung er-	
	ist <code>contract</code> nicht	gänzt	14
	<code>contractpars</code>	Umgebung definiert sich nicht	
		mehr selbst um	14
v0.5b		<code>ref:</code> neue Einstellungen	
	<code>juratitlepagebreak:</code> neue Op-	<code>nopar,</code> <code>nosentence,</code>	
	tion	<code>OnlyParagraph</code>	12
v0.5c		<code>\jura@@usetype:</code> neu (intern)	21
	<code>\getmaxpar: \#2@Paragraphformat</code>	<code>\jura@env@type:</code> neu (intern)	14
	expandiert zu seinem Argu-	<code>\jura@usetype:</code> neu (intern)	21

<code>\p@par: \jura@usetype</code> mit Argument <code>\jura@env@type</code> eingefügt	21	<code>\sentenceciteformat:</code> Argument verschoben	28
<code>\parciteformat:</code> Argument verschoben	27		