

KOMA-Script Datei `scrjura.dtx` *

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

2011/09/30

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 v0.6 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 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`, `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.

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 `\useshorthands` aus dem ohnehin bei deutschen Texten notwendigen `babel`-Paket:

```
\usepackage[...]{babel}
\useshorthands{'}
\defineshorthand{'S}{\Sentence\ignorespaces}
\defineshorthand{'.'}{. \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 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}
```

`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 Zähler wiederum wird erreicht, dass Paragraphen in das Inhaltsverzeichnis aufgenommen werden.

`juratoclevel`

```
14 \newcounter{juratoclevel}\setcounter{juratoclevel}{\@M}
15 \KOMA@key{juratotoc}[true]{%
16   \begingroup
17     \newif\if@setjuratoclevel\@setjuratocleveltrue
18     \let\juratoclevel@@FamilyUnknownKeyValue\FamilyUnknownKeyValue
19     \renewcommand*{\FamilyUnknownKeyValue}[4]{%
20       \setcounter{juratoclevel}{#1}%
21       \@setjuratoclevelfalse
22     }%
23     \KOMA@set@ifkey{juratoclevel}{@tempswa}{#1}%
24     \if@setjuratoclevel
25       \if@tempswa
```

```

26     \setcounter{juratoclevel}{2}%
27     \else
28     \setcounter{juratoclevel}{\@M}%
29     \fi
30     \fi
31 \endgroup
32 }

```

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

```

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

```

`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.

```

43 \KOMA@ifkey{juratitlepagebreak}{@juratitlepagebreak}

```

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

```

44 \newif\ifparnumber
45 \KOMA@key{parnumber}[true]{%
46   \ifstr{#1}{auto}{\AutoPar}{%
47     \ifstr{#1}{manual}{\ManualPar}{%
48       \def\FamilyElseValue{, 'auto', 'semi'}%
49       \KOMA@set@ifkey{parnumber}{parnumber}{#1}%
50     }%
51   }%
52 }

```

`paragraphmark` Mit diesen Optionen wird `\Paragraphmark` so umdefiniert, dass es entweder 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 werden!)

```

53 \newcommand*{\Paragraphmark}[1]{
54 \KOMACodeKey{paragraphmark}{%
55   \begingroup
56     \KOMACodeSet@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   \ifcase\number\@tempa
64     \endgroup
65     \let\Paragraphmark\@gobble
66   \or
67     \endgroup
68     \renewcommand*{\Paragraphmark}[1]{%
69       \markright{\csname MakeMarkcase\endcsname{##1}}}%
70   \or
71     \endgroup
72     \renewcommand*{\Paragraphmark}[1]{%
73       \markboth{\csname MakeMarkcase\endcsname{##1}}%
74         {\csname MakeMarkcase\endcsname{##1}}}%
75   \or
76     \endgroup
77     \renewcommand*{\Paragraphmark}[1]{%
78       \ifx
79         \@mkboth\@gobbletwo
80       \else
81         \markright{\csname MakeMarkcase\endcsname{##1}}%
82       \fi}%
83   \or
84     \endgroup
85     \renewcommand*{\Paragraphmark}[1]{%
86       \@mkboth{\csname MakeMarkcase\endcsname{##1}}%
87         {\csname MakeMarkcase\endcsname{##1}}}%
88   \else
89     \endgroup
90   \fi
91 }
92 \KOMACodeDeclareDeprecatedOption[scrjura]{markright}{paragraphmark=forceright}
93 \KOMACodeDeclareDeprecatedOption[scrjura]{markboth}{paragraphmark=forceboth}

```

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

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

```
94 \newcommand*\parcite@format{}{0}
95 \newcommand*\sentencecite@format{}{0}
```

Per Option kann die Voreinstellung verändert werden.

```
96 \KOMA@key{ref}{%
97   \begingroup
98     \KOMA@set@ncmdkey{ref}{@tempa}{%
99       {parlong}{1},{longpar}{1},{ParL}{1},%
100      {parshort}{2},{shortpar}{2},{ParS}{2},%
101      {parnumeric}{3},{numericpar}{3},{ParN}{3},%
102      {paroff}{4},{nopar}{4},%
103      {sentencelong}{10},{longsentence}{10},{SentenceL}{10},%
104      {sentenceshort}{20},{shortsentence}{20},{SentenceS}{20},%
105      {sentencenumeric}{30},{numericentence}{30},{SentenceN}{30},%
106      {sentenceoff}{40},{nosentence}{40},%
107      {long}{11},%
108      {short}{22},%
109      {numeric}{33},%
110      {paragraphonly}{44},{onlyparagraph}{44},%
111      {ParagraphOnly}{44},{OnlyParagraph}{44}%
112    }{#1}%
113    \@tempcnta=\@tempa\relax
114    \@tempcntb=\z@
115    \@whilenum \@tempcnta>9 \do{%
116      \advance\@tempcnta -10\relax
117      \advance\@tempcntb \@ne\relax
118    }%
119    \ifcase \@tempcnta
120    \or
121      \aftergroup\def\aftergroup\parcite@format
122      \aftergroup{\aftergroup0\aftergroup}%
123    \or
124      \aftergroup\def\aftergroup\parcite@format
125      \aftergroup{\aftergroup1\aftergroup}%
126    \or
127      \aftergroup\def\aftergroup\parcite@format
128      \aftergroup{\aftergroup2\aftergroup}%
129    \or
130      \aftergroup\def\aftergroup\parcite@format
131      \aftergroup{\aftergroup-\aftergroup1\aftergroup}%
132    \fi
133    \ifcase \@tempcntb
134    \or
135      \aftergroup\def\aftergroup\sentencecite@format
136      \aftergroup{\aftergroup0\aftergroup}%

```

```

137 \or
138 \aftergroup\def\aftergroup\sentencecite@format
139 \aftergroup{\aftergroup1\aftergroup}%
140 \or
141 \aftergroup\def\aftergroup\sentencecite@format
142 \aftergroup{\aftergroup2\aftergroup}%
143 \or
144 \aftergroup\def\aftergroup\sentencecite@format
145 \aftergroup{\aftergroup-\aftergroup1\aftergroup}%
146 \fi
147 \endgroup
148 }
149 \KOMADeclareDeprecatedOption[scrjura]{parcitename}{ref=parlong}
150 \KOMADeclareDeprecatedOption[scrjura]{sentencecitename}{ref=sentencelong}

```

Optionen ausführen.

```
151 \KOMAProcessOptions\relax
```

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

```
152 \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.

```

153 \newcommand*{\contract}{%
154 \ifx\jura@env@type\@empty
155 \let\@doendpe\contract@doendpe
156 \let\Paragraph\contract@paragraph
157 \let\c@Paragraph\c@contractParagraph
158 \edef\cl@Paragraph{\cl@Paragraph\cl@contractParagraph}%
159 \let\SubParagraph\contract@subparagraph
160 \let\c@SubParagraph\c@contractSubParagraph
161 \edef\cl@SubParagraph{\cl@SubParagraph\cl@contractSubParagraph}%
162 \let\Par\contract@everypar
163 \let\Sentence\contract@sentence
164 \renewcommand*{\jura@env@type}{contract}%
165 \aliaskomafont{Paragraph}{contract.Paragraph}%
166 \else
167 \PackageError{contract}{nested 'contract' detected}{%
168 You may not use a 'contract' environment inside\MessageBreak
169 a '\jura@env@type' environment or after loading\MessageBreak

```

```

170     package 'scrjura' with option '\jura@env@type'!}%
171 \fi
172 }
173 \let\if@jura@skiphyperref\iftrue
174 \let\cl@Paragraph\@empty
175 \let\cl@SubParagraph\@empty
176 \newcounter{contractParagraph}
177 \renewcommand*{\thecontractParagraph}{%
178   {\contract@Paragraphformat{\arabic{contractParagraph}}}}
179 \DeclareRobustCommand*{\contract@Paragraphformat}[1]{\paragraphformat{#1}}
180 \newcommand*{\paragraphformat}[1]{\S~#1}
181 \newcounter{contractSubParagraph}
182 \@addtoreset{SubParagraph}{Paragraph}
183 \renewcommand*{\thecontractSubParagraph}{%
184   {\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.

<code>title</code>	Der Titel, der Kolumnentitel und der Verzeichniseintrag des Paragraphen.
<code>head</code>	Der Titel setzt dabei zunächst auch die anderen beiden. Man kann die
<code>entry</code>	beiden aber auf leer setzen. Insbesondere aber kann man sie mit den <code>no-</code>
<code>tocentry</code>	Optionen auch abschalten.
<code>nohead</code>	
<code>noentry</code>	185 \define@key{contract}{title}{%
<code>notocentry</code>	186 \def\contract@title{#1}%
	187 \ifx\contract@entry\relax\def\contract@entry{\contract@title}\fi
	188 \ifx\contract@head\relax\def\contract@head{\contract@title}\fi
	189 }
	190 \define@key{contract}{entry}{%
	191 \PackageWarning{scrjura}{deprecated option 'entry'.\MessageBreak
	192 You should use option 'tocentry' instead of\MessageBreak
	193 option 'entry'%
	194 }%
	195 \def\contract@entry{#1}}
	196 \define@key{contract}{tocentry}{\def\contract@entry{#1}}
	197 \define@key{contract}{noentry}[]{%
	198 \PackageWarning{scrjura}{deprecated option 'noentry'.\MessageBreak
	199 You should use option 'notocentry' instead of\MessageBreak
	200 option 'noentry'%
	201 }%
	202 \let\contract@entry\relax}
	203 \define@key{contract}{notocentry}[]{\let\contract@entry\relax}
	204 \define@key{contract}{head}{\def\contract@head{#1}}
	205 \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.

```
206 \define@key{contract}{number}{\def\contract@number{#1}}
```

`\contract@preskip` `\contract@postskip` `preskip` `postskip` 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.

```
207 \newcommand*{\contract@preskip}{2\baselineskip}
208 \newcommand*{\contract@postskip}{\baselineskip}
209 \define@key{contract}{preskip}{\def\contract@preskip{#1}}
210 \define@key{contract}{postskip}{\def\contract@postskip{#1}}
```

`dummy` Über diese 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.

```
211 \newif\ifcontract@dummy
212 \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!

```
213 \newkomafont{contract.Paragraph}{\sffamily\bfseries\large}
214 \newcommand*{\contract@paragraph@font}{\usekomafont{contract.Paragraph}%
215 \@hangfrom}
```

```
@AbsParagraph
\theH@AbsParagraph 216 % \textsf{hyperref}-Code, der noch nicht getestet ist!
\theHParagraph 217 \newcounter{@AbsParagraph}
\theHSubParagraph 218 \newcommand*{\theH@AbsParagraph}{P-\arabic{@AbsParagraph}}
219 \newcommand*{\theHParagraph}{\theH@AbsParagraph}
220 \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.

```

221 \newcommand*{\contract@paragraph}[1]{%
222   \ManualPar\parnumbertrue
223   \let\contract@title\relax
224   \let\contract@entry\relax
225   \let\contract@head\relax
226   \let\contract@number\relax
227   \contract@dummysfalse
228   \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.

```

229   \ifcontract@dummys\else
230     \@afterindentfalse
231     \addvspace{\contract@preskip}%
232   \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.

```

233   \ifx\contract@number\relax
234     \let\p@Paragraph\@empty
235     \let\theParagraph\thecontractParagraph
236     \refstepcounter{Paragraph}%
237   \else
238     \begingroup
239       \let\@elt\@stpelt
240       \cl@Paragraph
241     \endgroup
242     \protected@edef\theParagraph{\S~\contract@number}%
243     \protected@edef\@currentlabel{\theParagraph}%
244   \fi
245   \stepcounter{@AbsParagraph}%
246   \begingroup\expandafter\expandafter\expandafter\endgroup
247   \expandafter\ifx\csname if@skiphyperref\endcsname\relax
248   \else
249     \expandafter\let\csname if@jura@skiphyperref\expandafter\endcsname
250     \csname if@skiphyperref\endcsname
251   \fi
252   \if@jura@skiphyperref\else
253     \hyper@refstepcounter{@AbsParagraph}%
254     \typeout{absolute Nummer: \the@AbsParagraph^^JLabel: '\@currentHref'}%

```

255 \fi

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

256 \let\theSubParagraph\theParagraph

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

```
257 \ifcontract@dummy\else
258   \begingroup
259     \if@juratitlepagebreak\else\interlinepenalty\@M\fi
260     \contract@paragraph@font \theParagraph
261     \ifx\contract@title\relax\else\enskip\contract@title\fi
262     \ifx\contract@entry\relax\else
263       \addcontentsline{toc}{cpar}{%
264         \protect\numberline{\theParagraph}\contract@entry
265       }%
266       \addcontentsline{cpar}{cpar}{%
267         \protect\numberline{\theParagraph}\contract@entry
268       }%
269     \fi
270     \ifx\contract@head\relax\else
271       \expandafter\Paragraphmark\expandafter{%
272         \expandafter\theSubParagraph\expandafter\enskip\contract@head}%
273     \fi
274     \par
275   \endgroup\nobreak\vskip\contract@postskip
```

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

```
276   \jura@afterheading{contract}%
277 \fi
278 }
```

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

```
279 \newcommand*{\contract@subparagraph}[1]{%
280   \ManualPar\parnumbertrue
281   \let\contract@title\relax
282   \let\contract@entry\relax
283   \let\contract@head\relax
284   \let\contract@number\relax
285   \contract@dummyfalse
286   \ifx\relax#1\relax\else\setkeys{contract}{#1}\fi
287   \ifcontract@dummy\else
288     \@afterindentfalse
```

```

289     \vskip\contract@preskip
290 \fi
291 \ifx\contract@number\relax
292     \let\p@SubParagraph\@empty
293     \let\theSubParagraph\thecontractSubParagraph
294     \refstepcounter{SubParagraph}%
295 \else
296     \begingroup
297         \let\@elt\@stpelt
298         \cl@SubParagraph
299     \endgroup
300     \protected@edef\theSubParagraph{\theParagraph\contract@number}%
301     \protected@edef\@currentlabel{\theSubParagraph}%
302 \fi
303 \stepcounter{@AbsParagraph}%
304 \begingroup\expandafter\expandafter\expandafter\endgroup
305 \expandafter\ifx\csname if@skiphyperref\endcsname\relax
306 \else
307     \expandafter\let\csname if@jura@skiphyperref\endcsname
308     \csname if@skiphyperref\endcsname
309 \fi
310 \if@jura@skiphyperref\else
311     \hyper@refstepcounter{@AbsParagraph}%
312     \typeout{absolute Nummer: \the@AbsParagraph^JLabel: '\@currentHref'}%
313 \fi
314 \ifcontract@dummy\else
315     \begingroup
316         \if@juratitlepagebreak\else\interlinepenalty\@M\fi
317         \contract@paragraph@font \theSubParagraph
318         \ifx\contract@title\relax\else\enskip\contract@title\fi
319         \ifx\contract@entry\relax\else
320             \addcontentsline{toc}{cpar}{%
321                 \protect\numberline{\theSubParagraph}\contract@entry
322             }%
323             \addcontentsline{cpar}{cpar}{%
324                 \protect\numberline{\theSubParagraph}\contract@entry
325             }%
326         \fi
327         \ifx\contract@head\relax\else
328             \expandafter\Paragraphmark\expandafter{%
329                 \expandafter\theSubParagraph\expandafter\enskip\contract@head}%
330         \fi
331         \par
332     \endgroup
333     \nobreak\vskip\contract@postskip
334     \jura@afterheading{contract}%
335 \fi
336 }

```

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

```
337 \newcommand*{\AutoPar}{%
338   \expandafter\let\expandafter\jura@everypar
339   \csname \jura@type @everypar\endcsname
340 }
341 \newcommand*{\ManualPar}{%
342   \let\jura@everypar\relax
343 }
```

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

```
344 \newcommand*{\jura@afterheading}[1]{%
345   \@nobreaktrue
346   \everypar{%
347     \if@nobreak
348       \@nobreakfalse
349       \clubpenalty \@M
350       \if@afterindent \else
351         {\setbox\z@\lastbox}%
352       \fi
353     \else
354       \clubpenalty \@clubpenalty
355       \everypar{%
356         \jura@everypar
357       }%
358     \fi
359   \jura@everypar
360 }%
361 \def\jura@type{#1}%
362 \AutoPar
363 }
```

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

```
364 \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.

```
365 \CheckCommand*\@doendpe{\@endpetrue
366   \def\par{\@restorepar\everypar{ }\par\@endpefalse}\everypar
367   {\setbox\z@\lastbox}\everypar{ }\@endpefalse}}
```

```

368 \newcommand*{\contract@endpe}{%
369   \@endpetrue
370   \def\par{%
371     \@restorepar\everypar{%
372       \csname jura@everypar\endcsname
373     }%
374     \par\@endpefalse
375   }%
376   \everypar{%
377     {\setbox\z@\lastbox}\everypar{%
378       \csname jura@everypar\endcsname
379     }%
380     \@endpefalse
381   }%
382 }

```

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

```

383 \newcommand*{\l@cpar}[2]{%
384   \ifnum\value{juratoclevel}>\value{tocdepth}\else
385     \scr@ifundefinedorrelax{bprot@dottedtocline}{%
386       \@dottedtocline
387     }{%
388       \bprot@dottedtocline
389     }{\value{juratoclevel}}{\cpar@indent}{\cpar@numberwidth}{#1}{#2}%
390   \fi
391 }

```

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

```

392 \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.

```

393 \DeclareRobustCommand*{\jura@usetype}[1]{\jura@@usetype{#1}}
394 \newcommand*{\jura@@usetype}[1]{ }

```

`\contract@everypar` `\jura@everpar` für Verträge.

`\ifparnumber` Der Schalter gibt an, ob überhaupt mit Absatznummern gearbeitet werden 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.

```

395 \newcounter{par}
396 \renewcommand*{\thepar}{\arabic{par}}
397 \newcommand*{\theHpar}{\theH@AbsParagraph-\Roman{par}}
398 \newcommand*{\parformat}{(\thepar)}
399 \renewcommand*{\p@par}{\jura@usetype{\jura@env@type}\theSubParagraph\jura@separat
400 \@addtoreset{par}{Paragraph}
401 \@addtoreset{par}{SubParagraph}

402 \newcommand*{\contract@everypar}{%
403   \ifparnumber
404     \refstepcounter{par}%
405     \refstepcounter{sentence}%
406     \begingroup
407       \protected@write\@auxout{\let\contract@Paragraphformat\@firstofone}{%
408         \string\newmaxpar{contract}{\theSubParagraph}{\arabic{par}}}%
409       \getmaxpar\@tempa{contract}{\theSubParagraph}%
410       \typeout{Stored max is \@tempa}%
411       \ifnum \@tempa>\@ne
412         \parformat\nobreakspace
413       \fi
414     \endgroup
415   \else
416     \setcounter{sentence}{0}\refstepcounter{sentence}%
417   \fi
418 }

```

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

```

\ref@L
419 \newcommand*{\refL}{\@ifstar {\ref@L*}{\ref@L{}}}
420 \newcommand*{\ref@L}[2]{%
421   \begingroup
422     \def\parcite@format{0}%
423     \let\sentencecite@format\parcite@format
424     \ref#1{#2}%
425   \endgroup
426 }

```

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

`\ref@S`

```

427 \newcommand*{\refS}{\@ifstar {\ref@S*}{\ref@S{}}}
428 \newcommand*{\ref@S}[2]{%
429   \begingroup
430   \def\parcite@format{1}%
431   \let\sentencecite@format\parcite@format
432   \ref#1{#2}%
433   \endgroup
434 }

```

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

```

\ref@N
435 \newcommand*{\refN}{\@ifstar {\ref@N*}{\ref@N{}}}
436 \newcommand*{\ref@N}[2]{%
437   \begingroup
438   \def\parcite@format{2}%
439   \let\sentencecite@format\parcite@format
440   \ref#1{#2}%
441   \endgroup
442 }

```

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

```

443 \newcommand*{\refParagraph}{%
444   \@ifstar {\ref@Paragraph*}{\ref@Paragraph{}}
445 }
446 \newcommand*{\ref@Paragraph}[2]{%
447   \expandafter\ifx\csname r@#2\endcsname\relax
448     \ref#1{#2}%
449   \else
450     \begingroup

```

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

```

451     \expandafter\expandafter\expandafter\expandafter
452     \expandafter\expandafter\expandafter\def
453     \expandafter\expandafter\expandafter\expandafter
454     \expandafter\expandafter\expandafter\@tempb
455     \expandafter\expandafter\expandafter\expandafter
456     \expandafter\expandafter\expandafter{%
457     \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%

```

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

```

458     \def\@tempc##1##2\@nil{##1}%
459     \let\jura@separator\@gobble
460     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
461     \csname r@#2\endcsname\noexpand\@nil}%

```


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

```
462     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
463     \@tempa\@nil}%
464     \let\@@protect\protect
465     \let\protect\noexpand
466     \expandafter\edef\csname r@#2\endcsname{\@tempa}\@tempb}%
467     \let\protect\@@protect
468     \ref#1{#2}%
469   \endgroup
470 \fi
471 }
```

`\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.

```
472 \newcommand*{\refParagraphN}{%
473   \@ifstar {\ref@ParagraphN*}{\ref@ParagraphN{}}
474 }
475 \newcommand*{\ref@ParagraphN}[2]{%
476   \begingroup
477   \let\paragraphformat\relax
478   \ref@Paragraph{#1}{#2}%
479   \endgroup
480 }
```

`\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.

```
481 \newcommand*{\refPar}{%
482   \@ifstar {\ref@Par*}{\ref@Par{}}
483 }
484 \newcommand*{\ref@Par}[2]{%
485   \expandafter\ifx\csname r@#2\endcsname\relax
486     \ref#1{#2}%
487   \else
488     \begingroup
```

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

```
489     \expandafter\expandafter\expandafter\expandafter
490     \expandafter\expandafter\expandafter\def
491     \expandafter\expandafter\expandafter\expandafter
492     \expandafter\expandafter\expandafter\@tempb
493     \expandafter\expandafter\expandafter\expandafter
494     \expandafter\expandafter\expandafter{%
495     \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%
```

Den ersten Teil der Referenz in \@tempa ablegen.

```
496 \def\@tempc##1##2\@nil{##1}%
497 \let\jura@separator\@gobble
498 \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
499 \csname r@#2\endcsname\@nil}%
```

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

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

```
\refParL
\ref@ParX
516 % Das Gleiche zwingend lang.
517 \newcommand*\refParL{%
518 \@ifstar {\ref@ParX0*}{\ref@ParX0{}}
519 }
520 \newcommand*\ref@ParX}[3]{%
521 \begingroup
522 \def\parcite@format{#1}%
523 \let\sentencecite@format\parcite@format
524 \ref@Par{#2}{#3}%
525 \endgroup
526 }
```

\refParS Das Gleiche zwingend kurz.

```
527 \newcommand*\refParS{%
528 \@ifstar {\ref@ParX1*}{\ref@ParX1{}}
529 }
```

\refParN Das Gleiche zwingend numerisch.

```
530 \newcommand*\refParN{%
```

```

531 \@ifstar {\ref@ParN2*}{\ref@ParN2{}}
532 }
533 \newcommand*{\ref@ParN}[2]{%
534   \@ifnextchar [%]
535     {\ref@@ParN{#1}{#2}}%
536     {\ref@ParX{#1}{#2}}%
537 }
538 \newcommand*{\ref@@ParN}{%
539 \def\ref@@ParN#1#2[#3]#4{%
540   \begingroup
541     \renewcommand*{\parnumericformat}[1]{%
542       \csname @#3\endcsname{\number #1\relax}%
543     }%
544     \ref@ParX{#1}{#2}{#4}%
545   \endgroup
546 }

```

`\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.

```

547 \newcommand*{\refSentence}{%
548   \@ifstar {\ref@Sentence*}{\ref@Sentence{}}
549 }
550 \newcommand*{\ref@Sentence}[2]{%
551   \expandafter\ifx\csname r@#2\endcsname\relax
552     \ref#1{#2}%
553   \else
554     \begingroup

```

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

```

555     \expandafter\expandafter\expandafter\expandafter
556     \expandafter\expandafter\expandafter\def
557     \expandafter\expandafter\expandafter\expandafter
558     \expandafter\expandafter\expandafter\@tempb
559     \expandafter\expandafter\expandafter\expandafter
560     \expandafter\expandafter\expandafter{%
561     \expandafter\expandafter\expandafter\@gobble\csname r@#2\endcsname}%

```

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

```

562     \def\@tempc##1##2\@nil{##1}%
563     \let\jura@separator\@gobble
564     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
565     \csname r@#2\endcsname\noexpand\@nil}%

```

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

```

566     \def\@tempc##1##2##3##4\@nil{##3}%
567     \protected@edef\@tempa{\expandafter\expandafter\expandafter\@tempc
568       \@tempa}{%
569       \protect\G@refundefinedtrue
570       \nfss@text{\reset@font\bfseries ??}%
571       \@latex@warning{Reference ‘#2’ on page \thepage \space
572         with undefined sentence number}%
573     }\noexpand\@nil}%
574     \let\@@protect\protect
575     \let\protect\noexpand
576     \expandafter\edef\csname r@#2\endcsname{\@tempa}\@tempb}%
577     \let\protect\@@protect
578     \ref#1{#2}%
579   \endgroup
580 \fi
581 }

```

\refSentenceL
\ref@SentenceX

```

582 % Das Gleiche zwingend lang.
583 \newcommand*\refSentenceL{%
584   \@ifstar {\ref@SentenceX0*}{\ref@SentenceX0{}}
585 }
586 \newcommand*\ref@SentenceX}[3]{%
587   \begingroup
588     \def\parcite@format{#1}%
589     \let\sentencecite@format\parcite@format
590     \ref@Sentence{#2}{#3}%
591   \endgroup
592 }

```

\refSentenceS Das Gleiche zwingend kurz.

```

593 \newcommand*\refSentenceS{%
594   \@ifstar {\ref@SentenceX1*}{\ref@SentenceX1{}}
595 }

```

\refSentenceN Das Gleiche zwingend nummerisch.

```

596 \newcommand*\refSentenceN{%
597   \@ifstar {\ref@SentenceX2*}{\ref@SentenceX2{}}
598 }

```

\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`

```

599 \newcounter{sentence}[par]
600 \renewcommand*\thesentence{\arabic{sentence}}
601 \newcommand*\theHsentence{\theHpar-\arabic{sentence}}
602 \renewcommand*\p@sentence{\expandafter\p@@sentence}
603 \newcommand*\p@cite[1]{\p@par{\par@cite{\theHpar}}%
604   \jura@separator{\nobreakspace}}{\sentence@cite{#1}}
605 \newcommand*\contract@sentence{%
606   \refstepcounter{sentence}%
607   \ensuremath{\mbox{\fontsize\sf@size\z@\selectfont\thesentence}}}%
608 }

```

`\parciteformat` Zitierstil für Absätze.

```

609 \DeclareRobustCommand*\par@cite[1]{\parciteformat{#1}}
610 \newcommand*\parciteformat[1]{%
611   \ifcase \parcite@format
612     \expandafter\parlongformat
613   \or
614     \expandafter\parshortformat
615   \or
616     \expandafter\parnumericformat
617   \else
618     \unskip\expandafter\@gobble
619   \fi
620   {#1}%
621 }

```

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

```

622 \DeclareRobustCommand*\sentence@cite[1]{\sentenceciteformat{#1}}
623 \newcommand*\sentenceciteformat[1]{%
624   \ifcase \sentencecite@format
625     \expandafter\sentencelongformat
626   \or
627     \expandafter\sentenceshortformat
628   \or
629     \expandafter\sentencenumericformat
630   \else
631     \unskip\expandafter\@gobble
632   \fi
633   {#1}%
634 }

```

```

\parlongformat   Formatierung in den sechs Formen.
\parshortformat
\parnumericformat 635 \newcommand*{\parlongformat}[1]{\parname-#1}
\sentencelongformat 636 \newcommand*{\parshortformat}[1]{\parshortname-#1}
\sentenceshortformat 637 \newcommand*{\parnumericformat}[1]{\@Roman{\number #1\relax}}
\sentencenumericformat 638 \newcommand*{\sentencelongformat}[1]{\sentencename-#1}
639 \newcommand*{\sentenceshortformat}[1]{\sentenceshortname-#1}
640 \newcommand*{\sentencenumericformat}[1]{\@arabic{\number #1\relax}.}

```

```

\parname   Der Name eines Absatzes und eines Satzes.
\parshortname
\sentencename
\sentenceshortname
\scrjura@lang@error
641 \newcommand*{\parname}{Paragraph}
642 \AtBeginDocument{%
643   \providecaptionname{german}\parname{Absatz}%
644   \providecaptionname{german}\parshortname{Abs.}%
645   \providecaptionname{german}\sentencename{Satz}%
646   \providecaptionname{german}\sentenceshortname{S.}%
647   \providecaptionname{ngerman}\parname{Absatz}%
648   \providecaptionname{ngerman}\parshortname{Abs.}%
649   \providecaptionname{ngerman}\sentencename{Satz}%
650   \providecaptionname{ngerman}\sentenceshortname{S.}%
651   \providecaptionname{austrian}\parname{Absatz}%
652   \providecaptionname{austrian}\parshortname{Abs.}%
653   \providecaptionname{austrian}\sentencename{Satz}%
654   \providecaptionname{austrian}\sentenceshortname{S.}%
655   \providecaptionname{naustrian}\parname{Absatz}%
656   \providecaptionname{naustrian}\parshortname{Abs.}%
657   \providecaptionname{naustrian}\sentencename{Satz}%
658   \providecaptionname{naustrian}\sentenceshortname{S.}%
659 }
660 \providecommand*{\parname}{\scrjura@lang@error{\parname}}
661 \providecommand*{\parshortname}{\scrjura@lang@error{\parshortname}}
662 \providecommand*{\sentencename}{\scrjura@lang@error{\sentencename}}
663 \providecommand*{\sentenceshortname}{\scrjura@lang@error{\sentenceshortname}}
664 \newcommand*{\scrjura@lang@error}[1]{%
665   \PackageErrorNoLine{scrjura}{%
666     current language not supported%
667   }{%
668     Currently scrjura only supports languages ‘german’,
669     ‘ngerman’, ‘austrian’, and ‘naustrian’.\MessageBreak
670     It seems, that you are using another language (maybe
671     \languagename).\MessageBreak
672     Because of this you have to define \string#1\space by
673     yourself!\MessageBreak
674     It would be nice if you’ll send your definitions to the author.%
675   }%
676   \textbf{??}%
677 }

```

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

```

678 \newcommand*{\newmaxpar}[3]{%
679   \begingroup
680     \protected@edef\@tempa{#2}\@onelevel@sanitize\@tempa
681     \expandafter\xdef\csname max@#1@\@tempa\endcsname{#3}%
682   \endgroup
683 }
684 \newcommand*{\getmaxpar}[3]{%
685   \begingroup
686     \expandafter\let\csname #2@Paragraphformat\endcsname\@firstofone
687     \protected@edef\@tempa{#3}%
688     \@onelevel@sanitize\@tempa
689     \expandafter\ifx \csname max@#2@\@tempa\endcsname\relax
690       \edef\@tempa{\endgroup\edef\noexpand#1{\expandafter\the\value{par}}}%
691     \else
692       \edef\@tempa{\endgroup
693         \edef\noexpand#1{\csname max@#2@\@tempa\endcsname}}%
694     \fi
695   \@tempa
696 }

```

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		
<code>@AbsParagraph</code> (Zähler)	<u>216</u>	<code>\cpar@indent</code>	<u>33</u>
<code>\@doendpe</code>	<u>365</u>	<code>\cpar@numberwidth</code>	<u>33</u>
	A		
<code>\AutoPar</code>	<i>10, 337</i>		
	C		
<code>\contract</code>	<u>153</u>		
<code>contract</code> (Umgebung)	<u>153</u>		
<code>contract</code> (Option)	<u>0, 11</u>		
<code>\contract@everypar</code>	<u>395</u>		
<code>\contract@paragraph</code>	<u>185</u>		
<code>\contract@paragraph@font</code>	<u>213</u>		
<code>\contract@postskip</code>	<u>207</u>		
<code>\contract@preskip</code>	<u>207</u>		
<code>\contract@sentence</code>	<u>599</u>		
<code>\contract@subparagraph</code>	<u>279</u>		
		D	
		<code>dummy</code> (Option)	<u>211</u>
		E	
		<code>entry</code> (Option)	<u>185</u>
		G	
		<code>\getmaxpar</code>	<i>10, 678</i>
		H	
		<code>head</code> (Option)	<u>185</u>
		I	
		<code>\if@juratitlepagebreak</code>	<u>43</u>
		<code>\if@juratotoc</code>	<u>14</u>
		<code>\ifcontract@dummy</code>	<u>211</u>

<code>\ifparnumber</code>	10, 395	<code>postskip</code>	0, 207
J		<code>preskip</code>	0, 207
<code>\jura@usetype</code>	393	<code>ref</code>	94
<code>\jura@afterheading</code>	344	<code>sentencecitename</code>	94
<code>\jura@env@type</code>	152	<code>title</code>	185
<code>\jura@everypar</code>	364	<code>tocentry</code>	185
<code>\jura@separator</code>	392	P	
<code>\jura@usetype</code>	393	<code>\p@par</code>	395
<code>juratitlepagebreak</code> (Option)	0, 43	<code>\p@sentence</code>	599
<code>juratocindent</code> (Option)	33	<code>\Par</code>	10
<code>\juratoclevel</code>	4	<code>\par</code>	10
<code>juratoclevel</code> (Zähler)	14	<code>par</code> (Zähler)	395
<code>juratocnumberwidth</code> (Option) ..	33	<code>\Paragraph</code>	5
<code>juratotoc</code> (Option)	0, 14	<code>\Paragraphmark</code>	53
L		<code>paragraphmark</code> (Option)	0, 53
<code>\l@cpair</code>	383	<code>\parcite@fromat</code>	94
M		<code>\parciteformat</code>	10, 609
<code>\ManualPar</code>	10, 337	<code>parcitename</code> (Option)	94
<code>markboth</code> (Option)	53	<code>\parformat</code>	10
<code>markright</code> (Option)	53	<code>\parlongformat</code>	635
N		<code>\parname</code>	10, 641
<code>\newmaxpar</code>	10, 678	<code>parnumber</code> (Option)	44
<code>noentry</code> (Option)	185	<code>\parnumberfalse</code>	9, 10
<code>nohead</code> (Option)	185	<code>\parnumbertrue</code>	9, 10
<code>notocentry</code> (Option)	185	<code>\parnumericformat</code>	635
<code>number</code> (Option)	206	<code>\parshortformat</code>	635
O		<code>\parshortname</code>	641
Optionen:		<code>postskip</code> (Option)	0, 207
<code>contract</code>	0, 11	<code>preskip</code> (Option)	0, 207
<code>dummy</code>	211	R	
<code>entry</code>	185	<code>ref</code> (Option)	94
<code>head</code>	185	<code>\ref@L</code>	419
<code>juratitlepagebreak</code>	0, 43	<code>\ref@N</code>	435
<code>juratocindent</code>	33	<code>\ref@Par</code>	481
<code>juratocnumberwidth</code>	33	<code>\ref@Paragraph</code>	443
<code>juratotoc</code>	0, 14	<code>\ref@ParagraphN</code>	472
<code>markboth</code>	53	<code>\ref@ParX</code>	516
<code>markright</code>	53	<code>\ref@S</code>	427
<code>noentry</code>	185	<code>\ref@Sentence</code>	547
<code>nohead</code>	185	<code>\ref@SentenceX</code>	582
<code>notocentry</code>	185	<code>\refL</code>	419
<code>number</code>	206	<code>\refN</code>	435
<code>paragraphmark</code>	0, 53	<code>\refPar</code>	10, 481
<code>parcitename</code>	94	<code>\refParagraph</code>	10, 443
<code>parnumber</code>	44	<code>\refParagraphN</code>	472
		<code>\refParL</code>	516
		<code>\refParN</code>	530
		<code>\refParS</code>	527

<code>\refS</code>	427	<code>\thecontractSubParagraph</code>	10
<code>\refSentence</code>	10, 547	<code>\theH@AbsParagraph</code>	216
<code>\refSentenceL</code>	582	<code>\theHpar</code>	395
<code>\refSentenceN</code>	596	<code>\theHParagraph</code>	216
<code>\refSentencesS</code>	593	<code>\theHsentence</code>	599
S			
<code>\scrjura@lang@error</code>	641	<code>\theHSubParagraph</code>	216
<code>\Sentence</code>	9	<code>\thepar</code>	10, 395
<code>\sentence</code>	10	<code>\thesentence</code>	10, 599
sentence (Zähler)	599	title (Option)	185
<code>\sentencecite@format</code>	94	tocentry (Option)	185
<code>\sentenceciteformat</code>	10, 622	U	
sentencecite (Option)	94	Umgebungen:	
<code>\sentencelongformat</code>	635	contract	153
<code>\sentencename</code>	10, 641	Z	
<code>\sentencenumericformat</code>	635	Zähler:	
<code>\sentenceshortformat</code>	635	@AbsParagraph	216
<code>\sentenceshortname</code>	641	juratoclevel	14
<code>\SubParagraph</code>	6	par	395
T			
<code>\thecontractParagraph</code>	10	sentence	599

Change History

0.5c	<code>\contract@everypar: \contract@Paragraphformat</code>	15	fehlende Fehlermeldung er-
	expandiert beim Schreiben zu		format
	seinem Argument	23	Umgebung definiert sich nicht
	erstes Argument von		mehr selbst um
	<code>\newmaxpar</code> und zweites Ar-		ref: neue Einstellungen
	gument von <code>\getmaxpar</code>		noper, nosentence,
	ist <code>contract</code> nicht		OnlyParagraph
	<code>contractpars</code>	23	<code>\jura@usetypes</code> : neu (intern) 22
v0.5b	<code>juratitlepagebreak</code> : neue Op-		<code>\jura@env@types</code> : neu (intern) 15
	tion	12	<code>\jura@usetypes</code> : neu (intern) . 22
v0.5c	<code>\getmaxpar: \#2@Paragraphformat</code>		<code>\p@par: \jura@usetypes</code> mit Ar-
	expandiert zu seinem Argu-		gument <code>\jura@env@types</code> ein-
	ment	31	gefügt
	<code>\protected@edef</code> durch		<code>\parciteformat</code> : Argument ver-
	<code>\edef</code> ersetzt	31	schoben
v0.5d	<code>\contract: \jura@env@types</code>		<code>\sentenceciteformat</code> : Argu-
	wird gesetzt	15	ment verschoben
			v0.5e
			<code>\Paragraphmark</code> : Beachtung von
			<code>\MakeMarkcase</code>
			v0.6
			<code>parnumber</code> : neue Option
			12

<code>\cpar@indent</code> : neu (intern) ..	12	Option verschoben	23
<code>\cpar@numberwidth</code> : neu (intern)	12	<code>entry</code> : Option veraltet	16
<code>\getmaxpar</code> : Argument 3 muss mit <code>\protected@edef</code> expandiert werden	31	<code>tocentry</code> : neue Option	16
<code>\if@juratotoc</code> : Schalter durch Zähler überflüssig	11	<code>noentry</code> : Option veraltet	16
<code>\ifparnumber</code> : Definition des Schalters in die Definition der		<code>notocentry</code> : neue Option	16
		<code>juratocindent</code> : neu	12
		<code>juratoclevel</code> : Zählerdefinition verschoben	11
		<code>juratocnumberwidth</code> : neu ...	12