

The `scripture` package

David Purton*

2026-06-12 v2.4

Abstract

The `scripture` package provides a set of macros for typesetting quotations from the Bible. It provides many features commonly seen in Bibles such as dropped text for chapter numbers, superscripts for verse numbers, indented lines for poetry sections, narrow sections and hanging paragraphs. A reference for the quotation can optionally be added.

Contents

1	Introduction	2
1.1	Bug reports and feature requests	2
1.2	Limitations	3
2	Documentation	3
2.1	Basic usage	3
2.2	Macros for typesetting scripture quotations	4
2.2.1	Functions	4
2.2.2	Environments	9
2.3	Provided hooks	12
2.4	Marks	13
2.5	Options	13
2.5.1	Options for the <code>center</code> environment	29
2.5.2	Options for the <code>flushright</code> environment	29
2.5.3	Options for the <code>hanging</code> environment	30
2.5.4	Options for the <code>midparachap</code> environment	31
2.5.5	Options for the <code>narrow</code> environment	32
2.5.6	Options for the <code>poetry</code> environment	32

*Email: dcpurton@marshwiggles.net

3	Implementation	34
3.1	Messages	34
3.2	Options	34
3.3	Hooks	47
3.4	Marks	47
3.5	The main <code>scripture</code> environment	48
3.5.1	Formatting and printing quotation references	51
3.5.2	Red lettering	61
3.5.3	Selecting language	62
3.5.4	Formatting and printing chapter numbers	62
3.5.5	Formatting and printing verse numbers	72
3.5.6	Printing pilcrows	74
3.5.7	Calculating the length of the final line in a paragraph	75
3.5.8	The <code>scripture</code> environment	76
3.6	Scripture internal environments	83
3.6.1	The <code>center</code> environment	85
3.6.2	The <code>flushright</code> environment	86
3.6.3	The <code>hanging</code> environment	88
3.6.4	The <code>midparachap</code> environment	95
3.6.5	The <code>narrow</code> environment	100
3.6.6	The <code>poetry</code> environment	102
3.7	Public L ^A T _E X Interface	113
	 Change History	 119
	 Index	 121

1 Introduction

Bibles tend to use a number of standard features when laying out chapters and verses. The main goal for the `scripture` package is to be able to quickly and easily input text and have it output nicely with minimal manual adjustment.

The package supports dropped text for chapter numbers, superscripts for verse numbers, appropriately indented lines for the first and second halves of poetry, flush right sections, centred sections, narrow sections and hanging sections.

The package could be used for typesetting an entire Bible, but it is mainly envisioned that it will be used for producing Scripture quotations. The passage reference can be included on the last line if there is space or failing that on the following line.

1.1 Bug reports and feature requests

Bug reports and feature requests can be made at the `scripture` package GitHub repository. See <https://github.com/dcpurton/scripture>.

1.2 Limitations

The `scripture` package makes extensive use of L^AT_EX `list` environments and the `\parshape` command. This means there are some limitations. In particular, you should not:

- Nest the `scripture` environment or any of its internal environments.
- Modify the `\parshape` within a `scripture` environment.
- Include any other `list` based environment within a `scripture` environment.
- Include any `\section` (or similar) functions within the `scripture` environment. A simple `\heading` function is provided instead.

The code which attempts to avoid drop chapters clashing with text is also a bit fragile and can easily break, especially when there are unexpected paragraph breaks.

Footnotes are supported, but as the `midparachap` environment is implemented with a `\vbox`, footnotes will vanish. In this case you must use `\footnotemark` and then `\footnotetext` after the `midparachap` environment.

The `poetry` environment uses `\obeylines`. This means that it can not be placed in the argument of a macro. If you want to use the `poetry` environment in a beamer slide, you must use the `fragile` option for the `beamer frame` environment.

Some attempt is made to encourage appropriate page breaks, especially just before a reference is added to a quotation, but this can still sometimes fail and a reference will end up on the following page. Usually a little more stretchable space on the page before the incorrect break is enough to obtain acceptable output.

At least version 2022-11-01 of the L^AT_EX format is required.

2 Documentation

2.1 Basic usage

Scripture quotations must be set within a `scripture` environment. A simple quotation might be produced as follows:

```
\begin{scripture}[Isaiah 56:1--2][version=NIVUK]
\ch{56}This is what the \name{Lord} says:
\begin{poetry}
  \Maintain justice
  and do what is right,

  for my salvation is close at hand
  and my righteousness will soon be revealed.

  \vs{2}Blessed is the one who does this---
  the person who holds it fast,

  who keeps the Sabbath without desecrating it,
  and keeps their hands from doing any evil.'
\end{poetry}
\end{scripture}
```

56 This is what the LORD says:
‘Maintain justice
and do what is right,
for my salvation is close at hand
and my righteousness will soon
be revealed.
²Blessed is the one who does this –
the person who holds it fast,
who keeps the Sabbath without
desecrating it,
and keeps their hands from
doing any evil.’
Isaiah 56:1–2 (NIVUK)

2.2 Macros for typesetting scripture quotations

2.2.1 Functions

`\added` `\added{⟨words⟩}`

New: 2022-11-15 Some Bible translations mark out words added for clarity not found in the original language (e.g., in italics). This can be done using the `\added` function. The font and format can be set using the `added/font` and `added/format` options.

```
\begin{scripture}[John 1:18][verse/font=\bfseries,
  verse/format=#1, verse/sep=0.5em, version=KJV]
\vs{18}No man hath seen God at any time, the
  only begotten Son, which is in the bosom of the
  Father, he hath declared \added{him}.
\end{scripture}
```

18 No man hath seen God at any time, the only begotten Son, which is in the bosom of the Father, he hath declared *him*. **John 1:18 (KJV)**

`\ch` `\ch[⟨letter⟩]{⟨chapter number⟩}`
`\ch*` `\ch*[⟨letter⟩]{⟨chapter number⟩}`
`\newch` `\newch[⟨letter⟩]`
`\newch*` `\newch*[⟨letter⟩]`

Format a chapter number. By default these are formatted using drop text taking up two lines and the text is separated from the chapter number by a horizontal space of 0.5em. The drop text is aligned to the top of `⟨letter⟩` which defaults to the value of `chapter/xchar`, which in turn defaults to a capital X in the default font.

If the `chapter/drop` option is `true` (default), then this function attempts to add sufficient space for the dropped chapter when its paragraph contains only one line. The starred version suppresses this behaviour.

`\newch` and `\newch*` behave similarly except that the chapter number is automatically incremented. The initial chapter can be set by passing `chapter` to the `scripture` environment.

```
\ch example
\begin{scripture}[Exodus 20:1--2]
\ch{20}And God spoke all these words:
\begin{hanging}
\noindent\vs{2}`I am the \name{Lord}
  your God, who brought you out of Egypt,
  out of the land of slavery.
\end{hanging}
\end{scripture}
```

```
\ch* example
\begin{scripture}[Exodus 20:1--2]
\ch*{20}And God spoke all these words:
\begin{hanging}
\nohang\vs{2}`I am the \name{Lord} your
  God, who brought you out of Egypt, out of
  the land of slavery.
\end{hanging}
\end{scripture}
```

```
\ch example
20 And God spoke all these words:
  2'I am the LORD your God, who
  brought you out of Egypt, out of the
  land of slavery.
Exodus 20:1-2
```

```
\ch* example
20 And God spoke all these words:
  2'I am the LORD your God, who
  brought you out of Egypt, out of the
  land of slavery.
Exodus 20:1-2
```

\extraskip \extraskip

Add a small vertical space of **\extraskip** between a paragraph. This function attempts to add additional space to avoid a clash with a drop chapter if required.

```
\begin{scripture}[Colossians 1:2]
\vs{2}To God's holy people in Colossae, the
faithful brothers and sisters in Christ:

\extraskip

Grace and peace to you from God our Father.
\end{scripture}
```

²To God's holy people in Colossae, the faithful brothers and sisters in Christ:

Grace and peace to you from God our Father. **Colossians 1:2**

\heading \heading{<section heading>}

New: 2024-11-19 The **\heading** function provides a simple section heading command for use within the **scripture** environment. Only one level is heading is provided. Further levels should be placed outside of the **scripture** environment. Additionally, the **\heading** function may not be used within a **scripture** inner environment.

The settings of headings can be controlled using the following options:

- **heading/aboveskip**
- **heading/afterindent**
- **heading/align**
- **heading/belowskip**
- **heading/font**
- **heading/format**
- **heading/hide**
- **heading/show**

Notes:

- **\section** and related functions are not supported within the **scripture** environment.
- The **\heading** function is ignored when the **inline** option is true.

```

\begin{scripture}[Luke 21:5-6]
\heading{Signs of the End of the Age}

\vs{5}Some of his disciples were remarking
about how the temple was adorned with beautiful
stones and with gifts dedicated to God. But
Jesus said, \vs{6}`As for what you see here,
the time will come when not one stone will be
left on another; every one of them will be
thrown down.'
\end{scripture}

```

Signs of the End of the Age

⁵Some of his disciples were remarking about how the temple was adorned with beautiful stones and with gifts dedicated to God. But Jesus said, ⁶“As for what you see here, the time will come when not one stone will be left on another; every one of them will be thrown down.” **Luke 21:5-6**

<code>\name</code>	<code>\name{<divine name>}</code>
<code>\LORD</code>	<code>\LORD</code>
<code>\GOD</code>	<code>\GOD</code>

New: 2022-09-01 Modern Bible translations frequently represent God’s personal divine name using a small capitals typeface. This can be done using the `\name` function. The font and format can be set using the `name/font` and `name/format` options. The `\LORD` and `\GOD` macros equivalent to `\name{Lord}` and `\name{God}` respectively.

Note: Watch out that trailing spaces are not eaten by `\LORD` and `\GOD`. Using `\name` is safer and semantically better.

```

\begin{scripture}[Genesis 15:1--2][version=ESV]
\ch{15}After these things the word of the
\name{Lord} came to Abram in a vision: ``Fear
not, Abram, I am your shield; your reward shall
be very great.'' \vs{2}But Abram said, ``O Lord
\name{God}, what will you give me, for I
continue childless, and the heir of my house is
Eliezer of Damascus?''
\end{scripture}

```

15 After these things the word of the LORD came to Abram in a vision: “Fear not, Abram, I am your shield; your reward shall be very great.” ²But Abram said, “O Lord GOD, what will you give me, for I continue childless, and the heir of my house is Eliezer of Damascus?”

Genesis 15:1-2 (ESV)

<code>\nofirstverse</code>	<code>\nofirstverse</code>
----------------------------	----------------------------

New: 2022-11-13 When `verse/first` is true, this causes the next `\vs` to print normally, rather than using the format specified by `verse/firstformat`. You might do this if your quote begins half way through a paragraph and the first verse should be printed normally.

<code>\nohang</code>	<code>\nohang</code>
----------------------	----------------------

In a `hanging` environment, this function indents the first line of the paragraph a length of `hanging/hang` + `parindent`. Remaining lines continue to be indented by a length of `hanging/hang`.

```

\begin{scripture}[Genesis 1:10--11]
\begin{hanging}
\noindent\vs{10}God called the dry ground
`land', and the gathered waters he called
`seas'. And God saw that it was good.

\nohang\vs{11}Then God said, `Let the land
produce vegetation: seed-bearing plants and
trees on the land that bear fruit with seed
in it, according to their various kinds.'
And it was so.
\end{hanging}
\end{scripture}

```

¹⁰God called the dry ground 'land', and the gathered waters he called 'seas'. And God saw that it was good.

¹¹Then God said, 'Let the land produce vegetation: seed-bearing plants and trees on the land that bear fruit with seed in it, according to their various kinds.' And it was so.

Genesis 1:10–11

`\nopilcrow` `\nopilcrow`

New: 2024-01-18

When `\pilcrow` is true, this causes the next pilcrow that would normally be printed to be suppressed. You might do this if your quote begins half way through a paragraph and so should not be marked with a pilcrow.

`\redletteron` `\redletteron`
`\redletteroff` `\redletteroff`

New: 2022-08-31

Use `\redletteron` and `\redletteroff` to mark the beginning and end of sections of text that should be attributed to Jesus speaking. Some attempt is made to ensure that the correct colour continues after an inner environment if `\redletteron` or `\redletteroff` occurs inside the inner environment. If you experience problems with unexpected colour, try to ensure that matching `\redletteron` and `\redletteroff` marks occur at the same group level. The colour of the text can be set using the `redletter/colour` option. These commands only have an effect when the `redletter` option is true.

```

\begin{scripture}[Matthew 21:16--17][redletter]
\vs{16}`Do you hear what these children are saying?'
they asked him.

\redletteron `Yes,' \redletteroff replied Jesus,
\redletteron `have you
never read,
\begin{poetry}
\thinspace ``From the lips of children and infants
you, Lord, have called forth your praise''?'
\end{poetry}\redletteroff

\vs{17}And he left them and went out of the city to
Bethany, where he spent the night.
\end{scripture}

```

¹⁶‘Do you hear what these children are saying?’ they asked him.

‘Yes,’ replied Jesus, ‘have you never read,

“From the lips of children and infants you, Lord, have called forth your praise”?’

¹⁷And he left them and went out of the city to Bethany, where he spent the night.

Matthew 21:16–17

`\scripturecurrentchapter` `\scripturecurrentchapter`
`\scripturecurrentverse` `\scripturecurrentverse`

New: 2022-08-07

`\scripturecurrentchapter` and `\scripturecurrentverse` globally hold the most recent values parsed to `\ch` and `\vs`. They can be initialised with `\renewcommand*` if required and could be used in hooks, for example.

<hr/> <code>\scripturesetup</code> <hr/>	<code>\scripturesetup{<options>}</code> Set default format for quotations. See section 2.5 for supported formatting <options>.
<hr/> <code>\scripturestyle</code> <code>\scripturestyle*</code> <hr/>	<code>\scripturestyle{<name>}{<options>}</code> <code>\scripturestyle*{<name>}{<options>}</code> New: 2023-12-27 Set <options> which are saved as style with <name>. These can later be set with the style option. The starred version appends <options> to an existing style rather than overwriting all options.
<hr/> <code>\selah</code> <hr/>	<code>\selah</code> New: 2022-07-29 In a poetry environment, this function inserts <i>Selah</i> at the end of the current line or the end of the following line if there is insufficient room. See options selah/font , selah/format , selah/sep and selah/text for options controlling the output.
<div> <pre> \begin{scripture}[Psalm 3:2][version=NIV84] \begin{poetry} \vs{2}Many are saying of me, ``God will not deliver him.``\selah \end{poetry} \end{scripture} </pre> </div> <div> ²Many are saying of me, “God will not deliver him.” <i>Selah</i> Psalm 3:2 (NIV84) </div>	
<hr/> <code>\textright</code> <hr/>	<code>\textright{<text>}</code> New: 2022-08-30 This function inserts <text> at the end of the current line or the end of the following line if there is insufficient room. The textright/sep option controls the minimum allowed space before the <text> to be right aligned.
<div> <pre> \begin{scripture}[Jeremiah 2:29] \begin{poetry} \vs{5}`Why do you bring charges against me? You have all rebelled against me,' \textright{declares the \name{Lord}.} \end{poetry} \end{scripture} </pre> </div> <div> ⁵`Why do you bring charges against me? You have all rebelled against me,' declares the LORD. Jeremiah 2:29 </div>	

`\textscripture` `\textscripture[⟨reference⟩][⟨options⟩]{⟨quotation⟩}`

New: 2023-01-07

`\textscripture` sets an inline scripture quotation. The quotation is surrounded by the values of `inline/begin` and `inline/end` which default to ``` and `'`, respectively. This function also sets the `compact` option and places the `⟨reference⟩` inline. The reference format can be controlled using the following options:

- `inline/reference/format`
- `inline/reference/sep`
- `inline/version/delim`
- `inline/version/format`

Other `⟨reference⟩` and `⟨version⟩` options are inherited.

This function is equivalent to using the `scripture` environment with the `inline` option.

Warning: `\par` is redefined to `\relax` inside the `\textscripture` function. If you have anything other than expected macros and environments inside your `\textscripture` function T_EX will likely get stuck in an infinite loop.

A famous verse in the Bible is:

```
\textscripture[John 3:16][verse/hide]{\vs{16}For
God so loved the world that he gave his one and
only Son, that whoever believes in him shall not
perish but have eternal life.}.
```

A famous verse in the Bible is: “For God so loved the world that he gave his one and only Son, that whoever believes in him shall not perish but have eternal life.” (**John 3:16**).

`\vs` `\vs{⟨verse number⟩}`

`\newvs` `\newvs`

Format a verse number. By default these are formatted using a superscript.

`\newvs` behaves similarly except that the verse number is automatically incremented. The initial verse can be set by passing `verse` to the `scripture` environment.

2.2.2 Environments

All scripture quotations must be set within a `scripture` environment. The rest of the environments documented below are only defined with the `scripture` environment. In most cases there is a default small `⟨aboveskip⟩` and `⟨belowskip⟩` inserted above and below these internal environments, but these are ignored if the environment is at the start or end of a `scripture` environment.

`center` `\begin{center}[⟨options⟩]`

`⟨centred text⟩`

`\end{center}`

Environment for typesetting centred sections of scripture quotations.

See section 2.5.1 for supported formatting `⟨options⟩`.

```
\begin{scripture}[Daniel 5:25]
\vs{25}`This is the inscription that was
written:
\begin{center}
\textsc{mene, mene, tekem, parsin}
\end{center}
\end{scripture}
```

²⁵‘This is the inscription that was written:

MENE, MENE, TEKEL, PARSIN
Daniel 5:25

```
flushright \begin{flushright}[\langle options \rangle]
\right aligned text
\end{flushright}
```

Environment for typesetting right aligned sections of scripture quotations.
See section 2.5.2 for supported formatting $\langle options \rangle$.

```
\begin{scripture}[Isaiah 40:5]
\begin{poetry}[belowskip = Opt]
\vs{5}And the glory of the \name{Lord}\ldots
and all people will see it together.
\end{poetry}

\begin{flushright}[aboveskip = Opt]
For the mouth of the \name{Lord} \\\
has spoken.'
\end{flushright}
\end{scripture}
```

⁵And the glory of the LORD...
and all people will see it
together.
For the mouth of the LORD
has spoken.’
Isaiah 40:5

```
hanging \begin{hanging}[\langle options \rangle]
\hanging text
\end{hanging}
```

Environment for typesetting hanging sections of scripture quotations.
See section 2.5.3 for supported formatting $\langle options \rangle$.

```
\begin{scripture}[Genesis 1:3--5]
\begin{hanging}
\vs{3}And God said, ‘Let there be light,’ and
there was light. \vs{4}God saw that the light
was good, and he separated the light from the
darkness. \vs{5}God called the light ‘day’,
and the darkness he called ‘night’. And there
was evening, and there was morn\~ing---the
first day.
\end{hanging}
\end{scripture}
```

³And God said, ‘Let there be light,’
and there was light. ⁴God
saw that the light was good,
and he separated the light from
the darkness. ⁵God called the
light ‘day’, and the darkness he
called ‘night’. And there was
evening, and there was morn-
ing—the first day.
Genesis 1:3–5

```
midparachap \begin{midparachap}[\langle options \rangle]
\text containing a chapter break
\end{midparachap}
```

Environment for typesetting paragraph with a chapter break within it.
See section 2.5.4 for supported formatting `<options>`.

```
\begin{scripture}[1 Corinthians 10:31--11:1]
\begin{midparachap}
\vs{31}So whether you eat or drink or
whatever you do, do it all for the glory of
God. \vs{32}Do not cause anyone to stumble,
whether Jews, Greeks or the church of
God---\vs{33}even as I try to please everyone
in every way. For I am not seeking my own
good but the good of many, so that they may
be saved. \ch{11}Follow my example, as I
follow the example of Christ.
\end{midparachap}
\end{scripture}
```

³¹So whether you eat or drink or whatever you do, do it all for the glory of God. ³²Do not cause anyone to stumble, whether Jews, Greeks or the church of God—³³even as I try to please everyone in every way. For I am not seeking my own good but the good of many, so that they may be saved. ¹Follow my example, as I follow the example of Christ.

1 Corinthians 10:31–11:1

```
narrow \begin{narrow}[<options>]
      <quotation>
\end{narrow}
```

Environment for typesetting narrow sections of scripture quotations. If there is no blank line before the start of a **narrow** environment the indent will be suppressed.

See section 2.5.5 for supported formatting `<options>`.

```
\begin{scripture}[2 Kings 18:18--19]
\vs{18}They called for the king; and Eliakim
son of Hilkiah the palace administrator, Shebna
the secretary, and Joah son of Asaph the
recorder went out to them.

\vs{19}The field commander said to them, `Tell
Hezekiah:

\begin{narrow}
\thinspace`This is what the great king, the
king of Assyria, says: on what are you basing
this confidence of yours?
\end{narrow}
\end{scripture}
```

¹⁸They called for the king; and Eliakim son of Hilkiah the palace administrator, Shebna the secretary, and Joah son of Asaph the recorder went out to them.

¹⁹The field commander said to them, ‘Tell Hezekiah:

‘“This is what the great king, the king of Assyria, says: on what are you basing this confidence of yours?”

2 Kings 18:18–19

```
poetry \begin{poetry}[<options>]
      <first half of verse>
      <second half of verse>

      <first half of verse>
      <second half of verse>
\end{poetry}
```

Environment for typesetting poetry sections of scripture quotations.

Poetry sections of the Bible often have a first half and a parallel second half which is indented a little. These should be set on separate consecutive lines. Leaving a blank line indicates that the next line should return to the first half indent.

See section 2.5.6 for supported formatting `<options>`.

<pre>\begin{scripture}[Isaiah 1:2] \begin{poetry} \vs{2}Hear me, you heavens! Listen, earth! For the \name{Lord} has spoken: 'I reared children and brought them up, but they have rebelled against me. \end{poetry} \end{scripture}</pre>	² Hear me, you heavens! Listen, earth! For the LORD has spoken: 'I reared children and brought them up, but they have rebelled against me. Isaiah 1:2
---	--

`scripture` `\begin{scripture}[<reference>][<options>]`
`<quotation>`
`\end{scripture}`

Main environment for typesetting scripture quotations.

The optional `<reference>` is printed at the end of the quotation. See section 2.5 for supported formatting `<options>`.

<pre>\begin{scripture}[John 3:16][version=NIV] \vs{16}For God so loved the world that he gave his one and only Son, that whoever believes in him shall not perish but have eternal life. \end{scripture}</pre>	¹⁶ For God so loved the world that he gave his one and only Son, that whoever believes in him shall not perish but have eternal life. John 3:16 (NIV)
--	--

2.3 Provided hooks

`scripture/chap/before`
`scripture/chap/after`
`scripture/verse/before`
`scripture/verse/after`

The following public hooks are available:

scripture/chap/before This hook is executed immediately before the `chapter/font` is set for the chapter.

scripture/chap/after This hook is executed immediately after the `chapter/sep` space is inserted.

scripture/verse/before This hook is executed immediately before the `verse/font` is set for the verse.

scripture/verse/after This hook is executed immediately after the `verse/sep` space is inserted.

New: 2022-08-07

2.4 Marks

scripture/heading The following marks are available and can be inserted using the new `\FirstMark`, `\TopMark` and `\LastMark` commands.

New: 2026-05-20

scripture/heading This mark is inserted in the `\heading` command.

scripture/verse This mark is inserted at the beginning of every verse. The format of the mark can be customised using the `mark/verse/format` option. **Note:** This mark is only inserted if the Bible book has been set with the `book` option.

The following example illustrates how these marks could be used to place the first and last verse on the page into the header.

```
\documentclass{article}
\usepackage{scripture}
\usepackage{fancyhdr}
\pagestyle{fancy}
\fancyhf{}
\lhead{\textbf{\FirstMark{scripture/verse}}}
\rhead{\textbf{\LastMark{scripture/verse}}}
\textbar\ \thepage}
\begin{document}
\begin{scripture}[] [noindent,book=Mark,chapter=1]
\vs{45}Instead he went out and began to talk
freely, spreading the news. As a result, Jesus
could no longer enter a town openly but stayed
outside in lonely places. Yet the people still
came to him from everywhere.

\extraskip

\ch{2}A few days later, when Jesus again
entered Capernaum, the people heard that he had
come home.
\end{scripture}
\end{document}
```

Mark 1:45 **Mark 2:1** | 1

⁴⁵Instead he went out and began to talk freely, spreading the news. As a result, Jesus could no longer enter a town openly but stayed outside in lonely places. Yet the people still came to him from everywhere.

2 A few days later, when Jesus again entered Capernaum, the people heard that he had come home.

2.5 Options

Default formatting `<options>` can be set using `\usepackage[<options>]{scripture}` or `\scripturesetup{<options>}`. Local formatting `<options>` can be set by passing them to the various environments described above.

aboveskip `aboveskip = <skip>` Default: Opt
`<skip>` inserted above the `scripture` environment.

added/font `added/font = ` Default: Empty

New: 2022-11-15 `` used for the `\added` function.

<hr/> added/format <hr/>	<code>added/format = $\langle macro \rangle$</code>	Default: <code>\emph{#1}</code>
New: 2022-11-15	$\langle macro \rangle$ used to format the words that should be highlighted as added by the translators using the <code>\added</code> function. #1 contains the text of the argument to <code>\added</code> .	
<hr/> after <hr/>	<code>after = $\langle token list \rangle$</code>	Default: <code>Empty</code>
New: 2024-06-01	Execute $\langle token list \rangle$ at the end of the <code>scripture</code> environment.	
<hr/> before <hr/>	<code>before = $\langle token list \rangle$</code>	Default: <code>Empty</code>
New: 2024-06-01	Execute $\langle token list \rangle$ at the beginning of the <code>scripture</code> environment.	
<hr/> belowskip <hr/>	<code>belowskip = $\langle skip \rangle$</code>	Default: <code>Opt</code>
	$\langle skip \rangle$ inserted below the <code>scripture</code> environment.	
<hr/> book <hr/>	<code>book = name</code>	Default: <code>Empty</code>
New: 2025-03-02	Initialise the book name for the current quotation. See also <code>chapter</code> and <code>verse</code> .	
<hr/> chapter <hr/>	<code>chapter = $\langle integer \rangle$</code>	Default: <code>1</code>
New: 2025-03-02	Initialise the chapter number when using automatic chapter numbering with <code>\newch</code> . See also <code>book</code> and <code>verse</code> .	
<hr/> chapter/aboveskip <hr/>	<code>chapter/aboveskip = $\langle skip \rangle$</code>	Default: <code>Opt</code>
New: 2026-06-01	$\langle skip \rangle$ inserted above a <code>\ch</code> when the <code>chapter/para</code> option is not set. See also <code>chapter/para/aboveskip</code> .	
<hr/> chapter/align <hr/>	<code>chapter/align = left right</code>	Default: <code>left</code>
New: 2024-12-30	When <code>chapter/align</code> is set to <code>right</code> a chapter will overlap to the left when <code>chapter/nodrop</code> is set. See also <code>chapter/valign</code> .	
<hr/> chapter/colour <hr/>	<code>chapter/colour = $\langle colour expression \rangle$</code>	Default: <code>.</code>
chapter/color <hr/>	The <code>l3color</code> $\langle colour expression \rangle$ to be used for chapter numbers. By default the current colour is inherited. <code>chapter/color</code> is an alias for <code>chapter/colour</code> .	
New: 2022-09-07		
<hr/> chapter/drop <hr/>	<code>chapter/drop = $\langle boolean \rangle$</code>	Default: <code>true</code>
	Whether the chapter should be set as a drop letter taking up two lines or as a normal letter taking up one line. The value of $\langle boolean \rangle$ is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>chapter/nodrop</code> .	
<hr/> chapter/font <hr/>	<code>chapter/font = $\langle font \rangle$</code>	Default: <code>\bfseries</code>
	$\langle font \rangle$ used for the <code>\ch</code> function.	

chapter/format	chapter/format = $\langle macro \rangle$	Default: #1
New: 2022-07-28	$\langle macro \rangle$ used to format the chapter for a scripture quotation. #1 contains the text of the chapter.	

chapter/hide	chapter/hide = $\langle boolean \rangle$	Default: false
New: 2022-08-15	Whether chapters should be shown or hidden. The value of $\langle boolean \rangle$ is optional and assumed to be true if not present. This option is the opposite of chapter/show . Note that the scripture/chap/before and scripture/chap/after hooks are still executed when chapters are hidden.	

<pre>\begin{scripture}[Romans 5:1--2][chapter/hide] \ch{5}Therefore, since we have been justified through faith, we have peace with God through our Lord Jesus Christ, \vs{2}through whom we have gained access by faith into this grace in which we now stand. And we boast in the hope of the glory of God. \end{scripture}</pre>	<p>¹Therefore, since we have been justified through faith, we have peace with God through our Lord Jesus Christ, ²through whom we have gained access by faith into this grace in which we now stand. And we boast in the hope of the glory of God.</p> <p style="text-align: right;">Romans 5:1–2</p>
---	--

chapter/language	chapter/language = $\langle language name \rangle$	Default: Empty
New: 2026-06-03	babel or polyglossia $\langle language name \rangle$ to be used for chapter numbers in case you want to use the main document language instead of the Bible text language.	

chapter/language/variant	chapter/language/variant = $\langle language variant name \rangle$	Default: Empty
New: 2026-06-03	polyglossia $\langle language variant name \rangle$ to be used for chapter numbers in case you want to use the main document language variant instead of the Bible text language variant.	

chapter/nodrop	chapter/nodrop = $\langle boolean \rangle$	Default: false
Whether the chapter should be set as a drop letter taking up two lines or as a normal letter taking up one line. The value of $\langle boolean \rangle$ is optional and assumed to be true if not present. This option is the opposite of chapter/drop .		

<pre>\begin{scripture}[John 1:1--3][chapter/nodrop] \ch{1}In the beginning was the Word, and the Word was with God, and the Word was God. \vs{2}He was with God in the beginning. \vs{3}Through him all things were made; without him nothing was made that has been made. \end{scripture}</pre>	<p>1 In the beginning was the Word, and the Word was with God, and the Word was God. ²He was with God in the beginning. ³Through him all things were made; without him nothing was made that has been made.</p> <p style="text-align: right;">John 1:1–3</p>
--	--

chapter/para	chapter/para = $\langle boolean \rangle$	Default: false
New: 2024-01-05	Set chapters in their own paragraph. The spacing before and after the chapter heading can be set using the chapter/para/aboveskip and chapter/para/belowskip options. The value of $\langle boolean \rangle$ is optional and assumed to be true if not present.	

```
\begin{scripture}[John 1:1][chapter/para,
chapter/format=CHAPTER~#1, version=KJV]
\ch{1}In the beginning was the Word, and the
Word was with God, and the Word was God.
\end{scripture}
```

CHAPTER 1

¹In the beginning was the Word,
and the Word was with God, and the
Word was God.

John 1:1 (KJV)

chapter/para/aboveskip `chapter/para/aboveskip = <skip>` Default: `\bigskipamount`

New: 2024-01-05 `<skip>` added above `\ch` when `chapter/para` is in effect. To be consistent with how the `parskip` package treats headings, no additional `\parskip` is inserted.

chapter/para/belowskip `chapter/para/belowskip = <skip>` Default: `\medskipamount`

New: 2024-01-05 `<skip>` added below `\ch` when `chapter/para` is in effect. To be consistent with how the `parskip` package treats headings, no additional `\parskip` is inserted.

chapter/para/indent `chapter/para/indent = <boolean>` Default: `true`

New: 2024-01-05 Whether to indent the line following a chapter heading when the `chapter/para` option is set to `true`. This only has an effect in the `scripture`, `narrow` and `midparachap` environments. The value of `<boolean>` is optional and assumed to be `true` if not present.

chapter/sep `chapter/sep = <dimen>` Default: `0.5em`

`<dimen>` used to separate a `\ch` from the following text.

chapter/show `chapter/show = <boolean>` Default: `true`

New: 2022-08-15 Whether chapters should be shown or hidden. The value of `<boolean>` is optional and assumed to be `true` if not present. This option is the opposite of `chapter/hide`.

chapter/showverse `chapter/showverse = <boolean>` Default: `false`

New: 2022-11-14 Whether `\vs{1}` should be automatically inserted after `\ch`. The value of `<boolean>` is optional and assumed to be `true` if not present. See also `midparachap/showverse` and `midparachap/hideverse`. Note that `chapter/showverse` take precedence over `midparachap/hideverse`.

```
\begin{scripture}[John 1:1][chapter/showverse]
\ch{1}In the beginning was the Word, and the
Word was with God, and the Word was God.
\end{scripture}
```

1 ¹In the beginning was the Word,
and the Word was with God, and
the Word was God. John 1:1

chapter/smash `chapter/smash = <boolean>` Default: `false`

New: 2024-12-30 Whether a chapter should be wrapped in `\smash` when `chapter/nodrop` is set. This allows the chapter number to be larger than the line height without affecting line spacing. The value of `<boolean>` is optional and assumed to be `true` if not present.

<code>chapter/valign</code>	<code>chapter/align = bottom middle top</code>	Default: <code>bottom</code>
New: 2024-12-30	Sets how a chapter will align vertically when <code>chapter/nodrop</code> is set.	

- `bottom`: Chapter will align at the text base line.
- `middle`: Chapter will align at the center of text.
- `top`: Chapter will align at the top of text.

See also `chapter/align`.

```

\documentclass{article}
\usepackage{babel}
\babelprovide[import, main]{english}
\babelprovide[import,
  onchar=ids fonts]{ancientgreek}
\babelfont[ancientgreek]{rm}[
  BoldFont=NewCM10-Bold]{NewCM10-Regular}
\usepackage{scripture}
\begin{document}
\begin{scripture}[Κατά Μάρκον 1:1--3][
  chapter/font=\huge\bfseries,
  chapter/align=right, chapter/nodrop,
  chapter/smash, chapter/valign=top,
  hanging/hang=1em, hanging/leftmargin=2em,
  hanging/verse/sep=1em, language=ancientgreek,
  verse/font=\sffamily\bfseries,
  verse/sep=0.25em, version=THGNT]
\begin{hanging}
\ch{1}Ἀρχὴ τοῦ εὐαγγελίου Ἰησοῦ χριστοῦ υἱοῦ
θεοῦ· \vs{2}καθὼς γέγραπται ἐν τῷ Ἡσαΐα τῷ
προφήτῃ· ἰδοὺ ἀποστέλλω τὸν ἄγγελόν μου πρὸ
προσώπου σου, ὃς κατασκευάσει τὴν ὁδόν σου.
\vs{3}φωνὴ βοῶντος ἐν τῇ ἐρήμῳ ἐτοιμάσατε τὴν
ὁδὸν κυρίου, εὐθείας ποιεῖτε τὰς τρίβους αὐτοῦ.
\end{hanging}
\end{scripture}
\end{document}

```

1 Ἀρχὴ τοῦ εὐαγγελίου Ἰησοῦ χρι-
στοῦ υἱοῦ θεοῦ· ²καθὼς γέγρα-
πται ἐν τῷ Ἡσαΐα τῷ προφήτῃ· ἰ-
δοὺ ἀποστέλλω τὸν ἄγγελόν μου
πρὸ προσώπου σου, ὃς κατασκευά-
σει τὴν ὁδόν σου. ³φωνὴ βοῶντος
ἐν τῇ ἐρήμῳ ἐτοιμάσατε τὴν ὁδὸν
κυρίου, εὐθείας ποιεῖτε τὰς τρίβους
αὐτοῦ.

Κατά Μάρκον 1:1-3 (THGNT)

<code>chapter/xchar</code>	<code>chapter/xchar = <character></code>	Default: <code>X</code>
New: 2025-01-01	Sets the character to which drop chapters and vertically aligned non-drop chapters are aligned. This can also be set for individual chapters using an optional argument to <code>\ch</code> .	

<code>colour</code>	<code>colour = <colour expression></code>	Default: <code>.</code>
<code>color</code>	The <code>l3color</code> <code><colour expression></code> to be used for the <code>scripture</code> environment. By default the current colour is inherited. <code>color</code> is an alias for <code>colour</code> .	
New: 2022-08-31		

compact	<code>compact = <boolean></code>	Default: <code>false</code>
----------------	--	-----------------------------

New: 2022-08-15	This option removes all paragraph breaks along with any <code>\extraskip</code> in the quotation so the whole quotation is a single paragraph of running prose. This option implies <code>chapter/nodrop</code> and all options relating to inner environments are ignored.
-----------------	---

The value of `<boolean>` is optional and assumed to be `true` if not present.

Warning: This option redefines `\par` to `\relax` inside the `scripture` environment. If you have anything other than expected macros and environments inside your `scripture` environment \TeX will likely get stuck in an infinite loop.

```
\begin{scripture}[Jonah 1:17--2:2a][compact]
\begin{midparachap}
  \vs{17}Now the \name{Lord} provided a huge
  fish to swallow Jonah, and Jonah was in the
  belly of the fish three days and three
  nights. \ch{2}From inside the fish Jonah
  prayed to the \name{Lord} his God. \vs{2}He
  said:
\end{midparachap}
\begin{poetry}
  `In my distress I called to the \name{Lord},
  and he answered me.
\end{poetry}
\end{scripture}
```

¹⁷Now the LORD provided a huge fish to swallow Jonah, and Jonah was in the belly of the fish three days and three nights. **2** From inside the fish Jonah prayed to the LORD his God. ²He said: ‘In my distress I called to the LORD, and he answered me.

Jonah 1:17–2:2a

defaults	<code>defaults</code>
-----------------	-----------------------

New: 2025-01-01	Resets options to defaults.
-----------------	-----------------------------

extraskip	<code>extraskip = <skip></code>	Default: <code>\medskipamount</code>
	<code><skip></code> used for the <code>\extraskip</code> function.	

font	<code>font = </code>	Default: <code>Empty</code>
	<code></code> used for the <code>scripture</code> environment.	

heading/aboveskip	<code>heading/aboveskip = <skip></code>	Default: <code>\bigskipamount</code>
New: 2024-11-19	<code><skip></code> inserted above a <code>\heading</code> .	

heading/afterindent	<code>heading/afterindent = <boolean></code>	Default: <code>false</code>
New: 2024-11-19	Whether to indent the first paragraph following a <code>\heading</code> . The value of <code><boolean></code> is optional and assumed to be <code>true</code> if not present.	

heading/align	<code>heading/align = left right center</code>	Default: <code>left</code>
New: 2024-11-19	Set the alignment of a <code>\heading</code> .	

heading/belowskip	<code>heading/belowskip = <skip></code>	Default: <code>\medskipamount</code>
New: 2024-11-19	<code><skip></code> inserted below a <code>\heading</code> .	

<hr/> <hr/>	heading/font	heading/font = $\langle font \rangle$	Default: <code>\small\itshape</code>
<hr/>	New: 2024-11-19	$\langle font \rangle$ used for a <code>\heading</code> .	
<hr/> <hr/>	heading/format	heading/format = $\langle macro \rangle$	Default: <code>#1</code>
<hr/>	New: 2024-11-19	$\langle macro \rangle$ used to format a <code>\heading</code> . <code>#1</code> contains the text of the heading.	
<hr/> <hr/>	heading/hide	heading/hide = $\langle boolean \rangle$	Default: <code>false</code>
<hr/>	New: 2024-11-19	Whether to hide <code>\heading</code> functions. The value of $\langle boolean \rangle$ is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>heading/show</code> .	
<hr/> <hr/>	heading/show	heading/show = $\langle boolean \rangle$	Default: <code>true</code>
<hr/>	New: 2024-11-19	Whether to show <code>\heading</code> functions. The value of $\langle boolean \rangle$ is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>heading/hide</code> .	
<hr/> <hr/>	indent	indent = $\langle boolean \rangle$	Default: <code>true</code>
<hr/>		Whether to indent the first line of a <code>scripture</code> environment or not by <code>parindent</code> . The value of $\langle boolean \rangle$ is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>noindent</code> .	
<hr/> <hr/>	inline	inline = $\langle boolean \rangle$	Default: <code>false</code>
<hr/>	New: 2023-01-07	Set the scripture quotation inline. The quotation is surrounded by the values of <code>inline/begin</code> and <code>inline/end</code> which default to <code>`</code> and <code>'</code> , respectively. This option also sets the <code>compact</code> option and sets the $\langle reference \rangle$ inline. The reference format can be controlled using the following options:	
		<ul style="list-style-type: none"> • <code>inline/reference/format</code> • <code>inline/reference/sep</code> • <code>inline/version/delim</code> • <code>inline/version/format</code> 	
		Other $\langle reference \rangle$ and $\langle version \rangle$ options are inherited.	
		Warning: <code>\par</code> is redefined to <code>\relax</code> when the <code>inline</code> option is set. If you have anything other than expected macros and environments inside your quotation \TeX will likely get stuck in an infinite loop.	
<hr/> <hr/>	inline/begin	inline/begin = $\langle text \rangle$	Default: <code>`</code>
<hr/>	New: 2023-01-07	Text inserted at the beginning of <code>\textscripture</code> or a <code>scripture</code> environment with the <code>inline</code> option set.	
<hr/> <hr/>	inline/end	inline/end = $\langle text \rangle$	Default: <code>'</code>
<hr/>	New: 2023-01-07	Text inserted between the end of <code>\textscripture</code> or a <code>scripture</code> environment with the <code>inline</code> option set and the $\langle reference \rangle$ (if specified).	

<hr/> inline/font <hr/>	<code>inline/font = </code>	Default: <i>Empty</i>
New: 2026-05-31	<code></code> used for <code>\textscripture</code> or a <code>scripture</code> environment with the <code>inline</code> option set.	

<hr/> inline/reference/format <hr/>	<code>inline/reference/format = <macro></code>	Default: <code>(#1)</code>
New: 2023-12-27	<code><macro></code> used to format the <code><reference></code> for an inline <code>scripture</code> quotation. <code>#1</code> contains the text of the <code><reference></code> and <code><version></code> (if specified).	

<hr/> inline/reference/sep <hr/>	<code>inline/reference/sep = <dimen></code>	Default: <code>0.5em</code>
New: 2023-12-27	The <code><dimen></code> between the end of an inline <code>scripture</code> quotation and its <code><reference></code> .	

<hr/> inline/version/delim <hr/>	<code>inline/version/delim = <delim></code>	Default: <code>{ }</code>
New: 2023-12-27	<code><delim></code> to use between the <code><reference></code> and the <code><version></code> of an inline <code>scripture</code> quotation.	

<hr/> inline/version/format <hr/>	<code>inline/version/format = <macro></code>	Default: <code>#1</code>
New: 2023-12-27	<code><macro></code> used to format the <code><version></code> for an inline <code>scripture</code> quotation. <code>#1</code> contains the text of the <code><version></code> .	

<hr/> language <hr/>	<code>language = <language name></code>	Default: <i>Empty</i>
New: 2022-11-02	<code>babel</code> or <code>polyglossia</code> <code><language name></code> to be used for the <code>scripture</code> environment. This calls <code>\selectlanguage{<language name>}</code> at the start of the environment. Make sure you have <code>babel</code> or <code>polyglossia</code> loaded and correctly set up for your specified language. If unset the current active language is used.	

Right-to-left languages are only supported with `babel` and `LuaLaTeX`. You will need to pass `bidirectional`, `layout=lists` as options to `babel` to ensure everything is laid out correctly.

```

\documentclass{article}
\usepackage[bidi=basic, layout=lists]{babel}
\babelprovide[import, main]{english}
\babelprovide[import, mapdigits]{persian}
\babelfont[persian]{rm}[
  Renderer=Harfbuzz]{Amiri}
\usepackage{scripture}
\begin{document}
\begin{scripture}[2--1:56]{إِشْعِيَا [version=NMV,
  language=persian]}
  \ch{56}خداوند چنین می‌فرماید:
  \begin{poetry}
    «انصاف را نگاه داشته،
    عدالت را به عمل آورید،
    زیرا نجات من نزدیک شده،
    و عدالت من به‌زودی منکشف خواهد شد.
    \vs{2}خوشا به حال آن که چنین کند،
    و کسی که بدین متمسک گردد؛
    که شَبَات را نگاه داشته، آن را بی‌حرمت نسازد،
    و دست خویش از هر عمل بد باز دارد.»
  \end{poetry}
\end{scripture}
\end{document}

```

۵۶ خداوند چنین می‌فرماید:

«انصاف را نگاه داشته،
عدالت را به عمل آورید،
زیرا نجات من نزدیک شده،
و عدالت من به‌زودی منکشف خواهد شد.
خوشا به حال آن که چنین کند،
و کسی که بدین متمسک گردد؛
که شَبَات را نگاه داشته، آن را بی‌حرمت نسازد،
و دست خویش از هر عمل بد باز دارد.»
(NMV) ۲-۱:۵۶ إِشْعِيَا

Note that for CJK scripts the `scripture` package is compatible with specialist packages like `luatexja` and `ctex`. These packages can give nicer kerning and protrusion out of the box. Although with some effort you can set up `babel` along with `microtype` to give comparable output.

```

\documentclass{article}
\usepackage{luatexja}
\usepackage{scripture}
\begin{document}
\begin{scripture}[マルコの福音書 9:33--37][
  version=NJB]
  \vs{33}一行はカペナウムに着いた。イエスは家に入ってから、弟子たちにお尋ねになった。「来る途中、何を論じ合っていたのですか。」
  \vs{34}彼らは黙っていた。来る途中、だれが一番偉いか論じ合っていたからである。
  \vs{35}イエスは腰を下ろすと、十二人を呼んで言われた。「だれでも先頭に立ちたいと思う者は、皆の後になり、皆に仕える者になりなさい。」
  \vs{36}それから、イエスは一人の子どもの手を取って、彼らの真ん中に立たせ、腕に抱いて彼らに言われた。
  \vs{37}「だれでも、このような子どもたちの一人を、わたしの名のゆえに受け入れる人は、わたしを受け入れるのです。また、だれでもわたしを受け入れる人は、わたしではなく、わたしを遣わされた方を受け入れるのです。」
\end{scripture}
\end{document}

```

³³一行はカペナウムに着いた。イエスは家に入ってから、弟子たちにお尋ねになった。「来る途中、何を論じ合っていたのですか。」³⁴彼らは黙っていた。来る途中、だれが一番偉いか論じ合っていたからである。³⁵イエスは腰を下ろすと、十二人を呼んで言われた。「だれでも先頭に立ちたいと思う者は、皆の後になり、皆に仕える者になりなさい。」³⁶それから、イエスは一人の子どもの手を取って、彼らの真ん中に立たせ、腕に抱いて彼らに言われた。³⁷「だれでも、このような子どもたちの一人を、わたしの名のゆえに受け入れる人は、わたしを受け入れるのです。また、だれでもわたしを受け入れる人は、わたしではなく、わたしを遣わされた方を受け入れるのです。」

マルコの福音書 9:33-37 (NJB)

<hr/> language/variant <hr/>	<code>language/variant = \langlelanguage variant name\rangle</code>	Default: <i>Empty</i>
<hr/> <small>New: 2022-11-02</small> <hr/>	polyglossia \langle language variant name \rangle to be used along with the specified language for the scripture environment. This inserts <code>\selectlanguage[\langlelanguage variant name\rangle]{\langlelanguage name\rangle}</code> at the start of the environment. Make sure you have polyglossia loaded and correctly set up for your specified language.	
<hr/> leftmargin <hr/>	<code>leftmargin = \langledimen\rangle</code>	Default: <i>Opt</i>
	\langle dimen \rangle used for the left margin of the scripture environment.	
<hr/> mark/verse/format <hr/>	<code>mark/verse/format = \langlemacro\rangle</code>	Default: <i>#1 #2:#3</i>
<hr/> <small>New: 2026-05-20</small> <hr/>	\langle macro \rangle used to format the scripture/verse mark. The arguments are:	
	#1 Current book name.	
	#2 Current chapter.	
	#3 Current verse.	
<hr/> name/font <hr/>	<code>name/font = \langlefont\rangle</code>	Default: <i>\scshape</i>
<hr/> <small>New: 2022-09-01</small> <hr/>	\langle font \rangle used for the \name function.	
<hr/> name/format <hr/>	<code>name/format = \langlemacro\rangle</code>	Default: <i>#1</i>
<hr/> <small>New: 2022-09-01</small> <hr/>	\langle macro \rangle used to format the divine name using the \name function. #1 contains the text of the argument to \name .	
<hr/> noindent <hr/>	<code>noindent = \langleboolean\rangle</code>	Default: <i>false</i>
	Whether to indent the first line of a scripture environment or not by parindent . The value of \langle boolean \rangle is optional and assumed to be true if not present. This option is the opposite of indent .	
	<pre>\begin{scripture}[1 Peter 3:18][noindent] \vs{18}For Christ also suffered once for sins, the righteous for the unrighteous, to bring you to God. He was put to death in the body but made alive in the Spirit. \end{scripture}</pre>	<p>¹⁸For Christ also suffered once for sins, the righteous for the unrighteous, to bring you to God. He was put to death in the body but made alive in the Spirit.</p> <p>1 Peter 3:18</p>
<hr/> parindent <hr/>	<code>parindent = \langledimen\rangle</code>	Default: <i>\parindent</i>
	\langle dimen \rangle to use when indenting the first line of a paragraph in the scripture environment.	
<hr/> parskip <hr/>	<code>parskip = \langleskip\rangle</code>	Default: <i>\parskip</i>
	\langle skip \rangle to use between paragraphs in the scripture environment.	

<hr/> <hr/> pilcrow	pilcrow = \langle boolean \rangle	Default: false
<hr/> New: 2024-01-18 <hr/>	Whether to insert a pilcrow (¶) at the beginning of each paragraph. This option only has an effect when using the compact and/or verse/para options. The value of \langle boolean \rangle is optional and assumed to be true if not present.	
	A subsequent pilcrow can be suppressed with the \nopilcrow macro. Inner environments start a new paragraph only if there is a blank line before them and similarly, there is only a new paragraph after an inner environment if it is followed by a blank line. New paragraphs are not inserted within a poetry environment, so to obtain a paragraph break within a block of poetry, use two poetry environments separated by a blank line. The exception to this is the midparachap environment which always starts and ends a paragraph.	
	<div><pre>\begin{scripture}[Deuteronomy 5:1][chapter/para, chapter/format=CHAPTER~#1, verse/para, verse/font=\bfseries, verse/format=#1, verse/sep=0.5em, pilcrow] \ch{5}Moses summoned all Israel and said: Hear, Israel, the decrees and the laws I declare in your hearing today. Learn them and be sure to follow them. \end{scripture}</pre></div>	<div>CHAPTER 5</div> <div>1 ¶ Moses summoned all Israel and said: ¶ Hear, Israel, the decrees and the laws I declare in your hearing today. Learn them and be sure to follow them. Deuteronomy 5:1</div>
<hr/> <hr/> pilcrow/sep	pilcrow/sep = \langle dimen \rangle	Default: 0.25em
<hr/> New: 2024-01-18 <hr/>	\langle dimen \rangle used to separate a pilcrow (¶) from the following text.	
<hr/> <hr/> redletter	redletter = \langle boolean \rangle	Default: false
<hr/> New: 2022-08-31 <hr/>	Whether to enable highlighting of Jesus' words in red. The value of \langle boolean \rangle is optional and assumed to be true if not present. The text needs to be marked up with \redletteron and \redletteroff for this option to have any effect.	
<hr/> <hr/> redletter/colour	redletter/colour = \langle colour expression \rangle	Default: red!80!black
<hr/> New: 2022-08-31 <hr/>	The l3color \langle colour expression \rangle to be used for the words of Jesus when the redletter option is true. \redletteron and \redletteroff must be used to mark up the text. redletter/color is an alias for redletter/colour.	
<hr/> <hr/> reference/align	reference/align = left right	Default: right
<hr/> New: 2023-01-08 <hr/>	Set the alignment of the reference. Note that in neither case will the reference break across lines.	
<hr/> <hr/> reference/colour	reference/colour = \langle colour expression \rangle	Default: .
<hr/> New: 2022-09-07 <hr/>	The l3color \langle colour expression \rangle to be used for the reference and version. By default the current colour is inherited. reference/color is an alias for reference/colour.	
<hr/> <hr/> reference/font	reference/font = \langle font \rangle	Default: \bfseries
	\langle font \rangle used when printing the \langle reference \rangle for a scripture quotation.	

<hr/> reference/format <hr/>	reference/format = $\langle macro \rangle$ $\langle macro \rangle$ used to format the $\langle reference \rangle$ for a scripture quotation. #1 contains the text of the $\langle reference \rangle$ and $\langle version \rangle$ (if specified).	Default: #1
<hr/> reference/language <hr/> New: 2026-06-03 <hr/>	reference/language = $\langle language name \rangle$ babel or polyglossia $\langle language name \rangle$ to be used for the reference in case you want to use the main document language instead of the Bible text language.	Default: Empty
<hr/> reference/language/variant <hr/> New: 2026-06-03 <hr/>	reference/language/variant = $\langle language variant name \rangle$ polyglossia $\langle language variant name \rangle$ to be used for the reference in case you want to use the main document language variant instead of the Bible text language variant.	Default: Empty
<hr/> reference/newline <hr/> New: 2023-01-8 <hr/>	reference/newline Always push the $\langle reference \rangle$ to a newline even if there is space for it to fit on the last line of the quotation. The option sets reference/sep to <code>\linewidth</code> so there is never room for the reference on the final line of the quotation.	
<hr/> reference/position <hr/> New: 2024-01-10 <hr/>	reference/position = start end Set the position of the reference at either the start or end of the quotation. If start is specified, then the reference will be set inline unless the quotation begins with an inner environment. Any \vs or \ch is suppressed at the start of the quotation in the main scripture environment.	Default: end
<div> <pre> \begin{scripture}[John 1:1][version=NIV, reference/position=start] \ch{1}In the beginning was the Word, and the Word was with God, and the Word was God. \end{scripture} </pre> </div> <div> John 1:1 (NIV) In the begin- ning was the Word, and the Word was with God, and the Word was God. </div>		
<hr/> reference/sep <hr/>	reference/sep = $\langle dimen \rangle$ The minimum $\langle dimen \rangle$ allowed between the end of a scripture quotation and its $\langle reference \rangle$ before the $\langle reference \rangle$ is pushed to the following line.	Default: 2em
<hr/> reference/start/inline <hr/> New: 2024-01-10 <hr/>	reference/start/inline = $\langle boolean \rangle$ When reference/position is set to start , this option controls whether the reference is set inline or on a line of its own before the quotation. Note that if the quotation begins with an inner environment the reference will always be on a new line unless the compact option is also set. The value of $\langle boolean \rangle$ is optional and assumed to be true if not present. This option is the opposite of reference/start/newline .	Default: true
<hr/> reference/start/newline <hr/> New: 2024-01-10 <hr/>	reference/start/newline = $\langle boolean \rangle$ When reference/position is set to start , this option controls whether the reference is set inline or on a line of its own before the quotation. Note that if the quotation begins with an inner environment the reference will always be on a new line unless the compact option is also set. The value of $\langle boolean \rangle$ is optional and assumed to be true if not present. This option is the opposite of reference/start/inline .	Default: false


```
\begin{scripture}[John 1:1]%
  [reference/position=start,
  reference/start/newline]
  \ch{1}In the beginning was the Word, and the
  Word was with God, and the Word was God.
\end{scripture}
```

John 1:1
1 In the beginning was the Word,
and the Word was with God, and
the Word was God.

reference/start/sep reference/start/sep = $\langle \text{dimen} \rangle$ Default: 1em

New: 2024-01-10 The space inserted between the reference and the quotation when **reference/position** is set to **start**. This only applies when the reference is inline and not on a line of its own.

reference/start/vsep reference/start/vsep = $\langle \text{skip} \rangle$ Default: Opt

New: 2024-01-10 The vertical skip inserted between the reference and the quotation when **reference/position** is set to **start**. This only applies when the reference is on a line of its own.

rightmargin rightmargin = $\langle \text{dimen} \rangle$ Default: Opt
 $\langle \text{dimen} \rangle$ used for the right margin of the **scripture** environment.

selah/font selah/font = $\langle \text{font} \rangle$ Default: \itshape
 $\langle \text{font} \rangle$ used when printing **\selah** in a **poetry** environment.

selah/format selah/format = $\langle \text{macro} \rangle$ Default: #1
 $\langle \text{macro} \rangle$ used to format the **\selah** in a **poetry** environment. #1 contains the value of **selah/text**.

selah/sep selah/sep = $\langle \text{dimen} \rangle$ Default: 1em
The minimum $\langle \text{dimen} \rangle$ allowed between the end of a poetry line and *Selah* before the *Selah* is pushed to the following line.

selah/text selah/text = $\langle \text{text} \rangle$ Default: Selah
 $\langle \text{text} \rangle$ used to for **\selah** in a **poetry** environment.

style style = $\langle \text{name} \rangle$
New: 2023-12-27 Set options previously saved with **\scripturestyle**. An error occurs if the style isn't defined.

textright/sep textright/sep = $\langle \text{dimen} \rangle$ Default: 1em
New: 2022-08-30 The minimum $\langle \text{dimen} \rangle$ allowed between the end of a line and text aligned to the right with **\textright** before the right aligned text is pushed to the following line.

verse verse = $\langle \text{integer} \rangle$ Default: 1
New: 2025-03-02 Initialise the chapter number when using automatic verse numbering with **\news**.
See also **book** and **chapter**.

verse/colour `verse/colour = <colour expression>` Default: .
verse/color The `\color` <colour expression> to be used for verse numbers. By default the current colour is inherited. `verse/color` is an alias for `verse/colour`.
New: 2022-09-07

verse/first `verse/first = <boolean>` Default: false
New: 2022-11-13 In prose sections, print the verse number in a different format if it is at the start of the paragraph. The format and separation between verse and text are controlled with the `verse/firstformat` and `verse/firstsep` options. Use `\nofirstverse` to suppress this at the next verse (e.g., if your quote starts mid-way though a paragraph and a normal verse should be printed). The value of <boolean> is optional and assumed to be true if not present.

```
\begin{scripture}[Luke 18:25--28][noindent,
  verse/first, version=NRSV]
\nofirstverse\vs{25}Indeed, it is easier for a
camel to go through the eye of a needle than for
someone who is rich to enter the kingdom of
God.''

\vs{26}Those who heard it said, ``Then who can
be saved?'' \vs{27}He replied, ``What is
impossible for mortals is possible for God.''

\vs{28}Then Peter said, ``Look, we have left our
homes and followed you.''
\end{scripture}
```

²⁵Indeed, it is easier for a camel to go through the eye of a needle than for someone who is rich to enter the kingdom of God.”

²⁶ Those who heard it said, “Then who can be saved?” ²⁷He replied, “What is impossible for mortals is possible for God.”

²⁸ Then Peter said, “Look, we have left our homes and followed you.”

Luke 18:25–28 (NRSV)

verse/firstformat `verse/firstformat = <macro>` Default: #1
New: 2022-11-13 <macro> used to format the first `\vs` in a paragraph. #1 contains the argument of `\vs`. Only used if `verse/first` is true.

verse/firstsep `verse/firstsep = <dimen>` Default: 0.5em
New: 2022-11-13 <dimen> used to separate the first `\vs` in a paragraph from the following text. Only used if `verse/first` is true.

verse/font `verse/font = ` Default: Empty
 used for the `\vs` function.

verse/format `verse/format = <macro>` Default: #1
 <macro> used to format the `\vs` function. #1 contains the argument of `\vs`.

verse/hide `verse/hide = <boolean>` Default: false
New: 2022-08-15 Whether verses should be shown or hidden. The value of <boolean> is optional and assumed to be true if not present. This option is the opposite of `verse/show`. Note that the `scripture/verse/before` and `scripture/verse/after` hooks are still executed when verses are hidden.

```

\begin{scripture}[Colossians 1:15--16][verse/hide]
\vs{15}The Son is the image of the invisible
God, the firstborn over all creation.
\vs{16}For in him all things were created:
things in heaven and on earth, visible and
invisible, whether thrones or powers or rulers
or authorities; all things have been created
through him and for him.
\end{scripture}

```

The Son is the image of the invisible God, the firstborn over all creation. For in him all things were created: things in heaven and on earth, visible and invisible, whether thrones or powers or rulers or authorities; all things have been created through him and for him.

Colossians 1:15–16

verse/language `verse/language = <language name>` **Default:** *Empty*

New: 2026-06-03 `babel` or `polyglossia` <language name> to be used for verse numbers in case you want to use the main document language instead of the Bible text language.

verse/language/variant `verse/language/variant = <language variant name>` **Default:** *Empty*

New: 2026-06-03 `polyglossia` <language variant name> to be used for verse numbers in case you want to use the main document language variant instead of the Bible text language variant.

verse/para `verse/para = <boolean>` **Default:** *false*

New: 2024-01-13 Whether to set each verse on its own line. The value of <boolean> is optional and assumed to be **true** if not present. Drop chapters and the shape of inner environments is preserved with this option unless the **compact** option is also set.

```

\begin{scripture}[Jonah 1:17--2:2a][verse/para,
verse/sep=0.5em, pilcrow, chapter/para,
chapter/format=CHAPTER-#1]
\begin{midparachap}
\vs{17}Now the \name{Lord} provided a huge
fish to swallow Jonah, and Jonah was in the
belly of the fish three days and three
nights. \ch{2}From inside the fish Jonah
prayed to the \name{Lord} his God. \vs{2}He
said:
\end{midparachap}

\begin{poetry}
`In my distress I called to the \name{Lord},
and he answered me.
\end{poetry}
\end{scripture}

```

17 ¶ Now the LORD provided a huge fish to swallow Jonah, and Jonah was in the belly of the fish three days and three nights.

CHAPTER 2

1 From inside the fish Jonah prayed to the LORD his God.

2 He said:

¶ ‘In my distress I called to the LORD,
and he answered me.

Jonah 1:17–2:2a

```

\begin{scripture}[Jonah 1:17--2:2a][verse/para,
verse/sep=0.5em, pilcrow, chapter/para,
chapter/format=CHAPTER-#1, compact]
\begin{midparachap}
\vs{17}Now the \name{Lord} provided a huge
fish to swallow Jonah, and Jonah was in the
belly of the fish three days and three
nights. \ch{2}From inside the fish Jonah
prayed to the \name{Lord} his God. \vs{2}He
said:
\end{midparachap}

\begin{poetry}
`In my distress I called to the \name{Lord},
and he answered me.
\end{poetry}
\end{scripture}

```

17 ¶ Now the LORD provided a huge fish to swallow Jonah, and Jonah was in the belly of the fish three days and three nights.

CHAPTER 2

1 From inside the fish Jonah prayed to the LORD his God.

2 He said: ¶ ‘In my distress I called to the LORD, and he answered me.

Jonah 1:17–2:2a

verse/presep `verse/presep = <skip>` Default: 0em

New: 2026-05-31 `<skip>` inserted before `\vs` when not at the beginning of a paragraph. Note that this can not be used with a custom `verse/firstformat` at the moment.

verse/sep `verse/sep = <dimen>` Default: 0.05em

New: 2022-07-28 `<dimen>` used to separate a `\vs` from the following text.

verse/show `verse/show = <boolean>` Default: true

New: 2022-08-15 Whether verses should be shown or hidden. The value of `<boolean>` is optional and assumed to be `true` if not present. This option is the opposite of `verse/hide`.

version `version = <version>` Default: Empty

`<version>` added to the `<reference>` for a `scripture` quotation.

version/delim `version/delim = <delim>` Default: { }

`<delim>` to use between the `<reference>` and the `<version>` of a `scripture` quotation.

version/format `version/format = <macro>` Default: (#1)

`<macro>` used to format the `<version>` for a `scripture` quotation. #1 contains the text of the `<version>`.

version/language `version/language = <language name>` Default: Empty

New: 2026-06-03 `babel` or `polyglossia` `<language name>` to be used for the version in case you want to use the main document language instead of the Bible text language.

version/language/variant `version/language/variant = <language variant name>` Default: Empty

New: 2026-06-03 `polyglossia` `<language variant name>` to be used for the version in case you want to use the main document language variant instead of the Bible text language variant.

<u>version/position</u>	version/position = withref end	Default: withref
New: 2024-01-10	Set the position of the version at either with the reference or at the end of the quotation if reference/position is set to start .	

```
\begin{scripture}[John 1:1][version=NIV,
  reference/position=start,
  version/position=end]
\ch{1}In the beginning was the Word, and the
Word was with God, and the Word was God.
\end{scripture}
```

John 1:1 In the beginning was
the Word, and the Word was with
God, and the Word was God.
(NIV)

2.5.1 Options for the **center** environment

If these options are passed directly to the **center** environment rather than to the package, **\scripturesetup** or the **scripture** environment, then the **center/** prefix should be dropped.

<u>center/aboveskip</u>	center/aboveskip = $\langle skip \rangle$	Default: $\backslash medskipamount$
	$\langle skip \rangle$ inserted above the center environment.	

<u>center/belowskip</u>	center/belowskip = $\langle skip \rangle$	Default: $\backslash medskipamount$
	$\langle skip \rangle$ inserted below the center environment.	

<u>center/leftmargin</u>	center/leftmargin = $\langle dimen \rangle$	Default: Opt
	$\langle dimen \rangle$ used for the left margin of the center environment.	

<u>center/rightmargin</u>	center/rightmargin = $\langle dimen \rangle$	Default: Opt
	$\langle dimen \rangle$ used for the right margin of the center environment.	

2.5.2 Options for the **flushright** environment

If these options are passed directly to the **flushright** environment rather than to the package, **\scripturesetup** or the **scripture** environment, then the **flushright/** prefix should be dropped.

<u>flushright/aboveskip</u>	flushright/aboveskip = $\langle skip \rangle$	Default: $\backslash medskipamount$
	$\langle skip \rangle$ inserted above the flushright environment.	

<u>flushright/belowskip</u>	flushright/belowskip = $\langle skip \rangle$	Default: $\backslash medskipamount$
	$\langle skip \rangle$ inserted below the flushright environment.	

<u>flushright/leftmargin</u>	flushright/leftmargin = $\langle dimen \rangle$	Default: Opt
	$\langle dimen \rangle$ used for the left margin of the flushright environment.	

<u>flushright/rightmargin</u>	flushright/rightmargin = $\langle dimen \rangle$	Default: Opt
	$\langle dimen \rangle$ used for the right margin of the flushright environment.	

2.5.3 Options for the `hanging` environment

If these options are passed directly to the `hanging` environment rather than to the package, `\scripturesetup` or the `scripture` environment, then the `hanging/` prefix should be dropped.

<hr/> <code>hanging/aboveskip</code> <hr/>	<code>hanging/aboveskip = <skip></code> <code><skip></code> inserted above the <code>hanging</code> environment.	Default: <code>\medskipamount</code>
<hr/> <code>hanging/belowskip</code> <hr/>	<code>hanging/belowskip = <skip></code> <code><skip></code> inserted below the <code>hanging</code> environment.	Default: <code>\medskipamount</code>
<hr/> <code>hanging/chapter/indent</code> <hr/>	<code>hanging/chapter/indent = <boolean></code> Indent the first line of chapters in <code>hanging</code> environments. If this is set to <code>false</code> , then the paragraph will be set has a normal hanging paragraph (if there is room).	Default: <code>true</code>
<hr/> <small>New: 2024-12-31</small> <hr/>		

```
\begin{scripture}[Genesis 1:1--3] [%
  hanging/chapter/indent=false]
\begin{hanging}
\ch{1}In the beginning God created the heavens
and the earth. \vs{2}Now the earth was
formless and empty, darkness was over the
surface of the deep, and the Spirit of God was
hovering over the waters.

\vs{3}And God said, ``Let there be light,''
and there was light.
\end{hanging}
\end{scripture}
```

1 In the beginning God created the heavens and the earth. ²Now the earth was formless and empty, darkness was over the surface of the deep, and the Spirit of God was hovering over the waters. ³And God said, “Let there be light,” and there was light.

Genesis 1:1–3

<hr/> <code>hanging/hang</code> <hr/>	<code>hanging/hang = <dimen></code> <code><dimen></code> used for the indent of subsequent lines of the <code>hanging</code> environment.	Default: <code>2em</code>
<hr/> <code>hanging/leftmargin</code> <hr/>	<code>hanging/leftmargin = <dimen></code> <code><dimen></code> used for the left margin of the <code>hanging</code> environment.	Default: <code>1em</code>
<hr/> <code>hanging/rightmargin</code> <hr/>	<code>hanging/rightmargin = <dimen></code> <code><dimen></code> used for the right margin of the <code>hanging</code> environment.	Default: <code>1em</code>
<hr/> <code>hanging/verse/left</code> <hr/>	<code>hanging/verse/left = <boolean></code> Align verse numbers in <code>hanging</code> environments at the left margin of the <code>hanging</code> environment. This is only used for verses at the start of a paragraph. This option is the opposite of <code>hanging/verse/right</code> .	Default: <code>false</code>
<hr/> <small>New: 2024-12-17</small> <hr/>		

```

\documentclass{article}
\usepackage{babel}
\babelprovide[import, main]{english}
\babelprovide[import,
  onchar=ids fonts]{ancientgreek}
\babelfont[ancientgreek]{rm}[
  BoldFont=NewCM10-Bold]{NewCM10-Regular}
\usepackage{scripture}
\begin{document}
\begin{scripture}[Κατά Μάρκον 1:28--30][
  hanging/hang=1em, hanging/leftmargin=2em,
  hanging/verse/left, hanging/verse/sep=1em,
  language=ancientgreek, verse/sep=0.25em,
  verse/font=\sffamily\bfseries, version=THGNT]
\begin{hanging}
\vs{28}Καὶ ἐξῆλθεν ἡ ἀκοὴ αὐτοῦ εὐθὺς πανταχοῦ
εἰς ὅλην τὴν περίχωρον τῆς Γαλιλαίας.

\vs{29}Καὶ εὐθὺς ἐκ τῆς συναγωγῆς ἐξελθόντες
ἦλθον εἰς τὴν οἰκίαν Σίμωνος καὶ Ἀνδρέου μετὰ
Ἰακώβου καὶ Ἰωάννου. \vs{30}ἡ δὲ πενθερὰ
Σίμωνος κατέκειτο πυρέσσουσα, καὶ εὐθὺς
λέγουσιν αὐτῇ περὶ αὐτῆς.
\end{hanging}
\end{scripture}
\end{document}

```

28 Καὶ ἐξῆλθεν ἡ ἀκοὴ αὐτοῦ εὐθὺς πανταχοῦ εἰς ὅλην τὴν περίχωρον τῆς Γαλιλαίας.

29 Καὶ εὐθὺς ἐκ τῆς συναγωγῆς ἐξελθόντες ἦλθον εἰς τὴν οἰκίαν Σίμωνος καὶ Ἀνδρέου μετὰ Ἰακώβου καὶ Ἰωάννου. 30 ἡ δὲ πενθερὰ Σίμωνος κατέκειτο πυρέσσουσα, καὶ εὐθὺς λέγουσιν αὐτῇ περὶ αὐτῆς.

Κατά Μάρκον 1:28–30 (THGNT)

<code>hanging/verse/right</code>	<code>hanging/verse/right = <boolean></code>	Default: true
----------------------------------	--	---------------

New: 2024-12-17

Align verse numbers in **hanging** environments at the left margin of the **hanging** environment. The verse numbers overlap to the left. This is only used for verses at the start of a paragraph. This option is the opposite of **hanging/verse/left**.

<code>hanging/verse/sep</code>	<code>hanging/verse/sep = <dimen></code>	Default: 0.05em
--------------------------------	--	-----------------

New: 2024-12-17

`<dimen>` used to separate a `\vs` from the following text in a **hanging** environment. This is only used for verses at the start of a paragraph.

2.5.4 Options for the **midparachap** environment

If these options are passed directly to the **midparachap** environment rather than to the package, `\scripturesetup` or the **scripture** environment, then the **midparachap/** prefix should be dropped.

<code>midparachap/hideverse</code>	<code>midparachap/hideverse = <boolean></code>	Default: false
------------------------------------	--	----------------

Whether a `\vs{1}` should automatically be inserted after a `\ch` when the `\ch` occurs mid-paragraph. This also affects `\ch` in **poetry** environments. The value of `<boolean>` is optional and assumed to be **true** if not present. This option is the opposite of **midparachap/showverse**. See also **chapter/showverse**.

```

\begin{scripture}[Hosea 11:12--12:1]%
  [midparachap/hideverse]
\begin{poetry}
  \vs{12}Ephraim has surrounded me with lies,
  Israel with deceit.

  And Judah is unruly against God,
  even against the faithful Holy One.

  \ch{12}Ephraim feeds on the wind;
  he pursues the east wind all day
  and multiplies lies and violence.

  He makes a treaty with Assyria
  and sends olive oil to Egypt.
\end{poetry}
\end{scripture}

```

¹²Ephraim has surrounded me with
lies,
Israel with deceit.
And Judah is unruly against God,
even against the faithful Holy
One.
12 Ephraim feeds on the wind;
he pursues the east wind all
day
and multiplies lies and violence.
He makes a treaty with Assyria
and sends olive oil to Egypt.
Hosea 11:12–12:1

<code>midparachap/showverse</code>	<code>midparachap/showverse = <boolean></code>	Default: <code>true</code>
------------------------------------	--	----------------------------

Whether a `\vs{1}` should automatically be inserted after a `\ch` when the `\ch` occurs mid-paragraph. This also affects `\ch` in `poetry` environments. The value of `<boolean>` is optional and assumed to be `true` if not present. This option is the opposite of `midparachap/hideverse`. See also `chapter/showverse`.

2.5.5 Options for the `narrow` environment

If these options are passed directly to the `narrow` environment rather than to the package, `\scripturesetup` or the `scripture` environment, then the `narrow/` prefix should be dropped.

<code>narrow/aboveskip</code>	<code>narrow/aboveskip = <skip></code>	Default: <code>\medskipamount</code>
-------------------------------	--	--------------------------------------

`<skip>` inserted above the `narrow` environment.

<code>narrow/belowskip</code>	<code>narrow/belowskip = <skip></code>	Default: <code>\medskipamount</code>
-------------------------------	--	--------------------------------------

`<skip>` inserted below the `narrow` environment.

<code>narrow/leftmargin</code>	<code>narrow/leftmargin = <dimen></code>	Default: <code>1em</code>
--------------------------------	--	---------------------------

`<dimen>` used for the left margin of the `narrow` environment.

<code>narrow/rightmargin</code>	<code>narrow/rightmargin = <dimen></code>	Default: <code>1em</code>
---------------------------------	---	---------------------------

`<dimen>` used for the right margin of the `narrow` environment.

2.5.6 Options for the `poetry` environment

If these options are passed directly to the `poetry` environment rather than to the package, `\scripturesetup` or the `scripture` environment, then the `poetry/` prefix should be dropped.

<hr/> <code>poetry/aboveskip</code> <hr/>	<code>poetry/aboveskip = <skip></code> <code><skip></code> inserted above the poetry environment.	Default: <code>\medskipamount</code>
<hr/> <code>poetry/belowskip</code> <hr/>	<code>poetry/belowskip = <skip></code> <code><skip></code> inserted below the poetry environment.	Default: <code>\medskipamount</code>
<hr/> <code>poetry/bigindent</code> <hr/>	<code>poetry/bigindent = <dimen></code> <code><dimen></code> used for the indent subsequent lines of a line of poetry in a poetry environment if it needs to flow on to the following line.	Default: <code>4em</code>
<hr/> <code>poetry/indent</code> <hr/>	<code>poetry/indent = <dimen></code> <code><dimen></code> used for the second half of a line of poetry in a poetry environment.	Default: <code>1em</code>
<hr/> <code>poetry/leftmargin</code> <hr/>	<code>poetry/leftmargin = <dimen></code> <code><dimen></code> used for the left margin of the poetry environment.	Default: <code>1em</code>
<hr/> <code>poetry/rightmargin</code> <hr/>	<code>poetry/rightmargin = <dimen></code> <code><dimen></code> used for the right margin of the poetry environment.	Default: <code>Opt</code>
<hr/> <code>poetry/verse/left</code> <hr/>	<code>poetry/verse/left = <boolean></code> Align verse numbers in poetry environments at the left margin of the scripture environment. This option is the opposite of poetry/verse/right .	Default: <code>false</code>

New: 2022-07-31

```
\begin{scripture}[Acts 15:15--18]%
  [poetry/leftmargin=2em, poetry/verse/left]
  \vs{15}The words of the prophets are in
  agreement with this, as it is written:
  \begin{poetry}
    \vs{16}`\thinspace`After this I will return
    and rebuild David's fallen tent.

    Its ruins I will rebuild,
    and I will restore it,

    \vs{17}that the rest of mankind may seek the Lord,
    even all the Gentiles who bear my name,

    says the Lord, who does these things' '---
    \vs{18}things known from long ago.
  \end{poetry}
\end{scripture}
```

15 The words of the prophets
are in agreement with this, as it
is written:

16 “After this I will return
and rebuild David’s fallen
tent.
Its ruins I will rebuild,
and I will restore it,
17 that the rest of mankind
may seek the Lord,
even all the Gentiles who
bear my name,
says the Lord, who does
these things”—
18 things known from long
ago.

Acts 15:15–18

<hr/> <code>poetry/verse/right</code> <hr/>	<code>poetry/verse/right = <boolean></code> Align verse numbers in poetry environments at the left margin of the poetry environment. The verse numbers overlap to the left. This option is the opposite of poetry/verse/left .	Default: <code>true</code>
<hr/> <code>poetry/verse/sep</code> <hr/>	<code>poetry/verse/sep = <dimen></code> <code><dimen></code> used to separate a <code>\vs</code> from the following text in a poetry environment. This is only used for verses at the start of a line.	Default: <code>0.05em</code>

New: 2022-07-31

New: 2022-07-31

3 Implementation

```

1 %<*package>
2 %<@@=scripture>
3 \NeedsTeXFormat{LaTeX2e}[2022-11-01]
4 \ProvidesExplPackage{scripture}{2026-06-12}{2.4}
5 {Format Scripture Quotations (DCP)}

```

3.1 Messages

`scripture` environments along with all internal environments can not be nested.

```

6 \msg_new:nnnn { scripture } { nested-environment }
7 { Nesting-of-scripture-environments-not-supported. }
8 { You-have-trying-to-nest-a-scripture-environment-or \\
9   one-of-its-internal-environments,~which-is-not-supported. }

```

Make sure `polyglossia` languages and variants are set up.

```

10 \msg_new:nnnn { scripture } { language-error }
11 { Language~`#1'~not-set-up. }
12 { Language~`#1'~has-not-been-set-up.~Make-sure-you-set-up-languages~
13   before-using-the-language-option-of-the-scripture-environment. }

```

Warn if a `language` is requested and neither `babel` nor `polyglossia` is loaded.

```

14 \msg_new:nnnn { scripture } { language-warning }
15 { Language-requested-without-support. }
16 { You-have-requested-language=#1,~but-neither-babel-nor-polyglossia~
17   have-been-loaded-and-set-up.~Ignoring-option-language=#1. }

```

Error if an `\heading` is not used in the main `scripture` environment.

```

18 \msg_new:nnnn { scripture } { heading-error }
19 { \token_to_str:N \heading \c_space_tl not-support-within-inner~
20   environments. }
21 { You-have-trying-to-use~\token_to_str:N \heading \c_space_tl within-the~
22   `#1'-environment,~but-this-macro-is-only-supported-in-the-main~
23   `scripture'-environment. }

```

Error if an unknown `style` is requested.

```

24 \msg_new:nnnn { scripture } { unknown-style }
25 { Style~#1~not-defined. }
26 { You-have-requested-style=#1,~but-it-has-not-been-defined.~
27   You-can-define-it-using~\token_to_str:N \scripturestyle.~
28   Ignoring-option-style=#1. }

```

3.2 Options

Key-value settings for the `scripture` environment and its inner environments.

```

29 \keys_define:nn { scripture }
30 {
31   , aboveskip .tl_set:N = \l__scripture_aboveskip_tl
32   , aboveskip .value_required:n = true
33   , aboveskip .initial:n = \c_zero_skip
34   , added / font .tl_set:N = \l__scripture_added_font_tl
35   , added / font .value_required:n = true
36   , added / font .initial:n =

```

```

37 , added / format .cs_set:Np = \__scripture_added_format:n #1
38 , added / format .value_required:n = true
39 , added / format .initial:n = \emph {#1}
40 , after .tl_set:N = \l__scripture_after_tl
41 , after .value_required:n = true
42 , after .initial:n =
43 , before .tl_set:N = \l__scripture_before_tl
44 , before .value_required:n = true
45 , before .initial:n =
46 , belowskip .tl_set:N = \l__scripture_belowskip_tl
47 , belowskip .value_required:n = true
48 , belowskip .initial:n = \c_zero_skip
49 , book .tl_gset:N = \g__scripture_book_tl
50 , book .value_required:n = true
51 , book .initial:n =
52 , colour .tl_set:N = \l__scripture_colour_tl
53 , colour .value_required:n = true
54 , colour .initial:n = .
55 , color .meta:n = { colour = {#1} }
56 , compact .bool_set:N = \l__scripture_compact_bool
57 , compact .default:n = true
58 , compact .initial:n = false
59 , chapter .int_gset:N = \g__scripture_chapter_int
60 , chapter .value_required:n = true
61 , chapter .initial:n = \c_one_int
62 , chapter / aboveskip .tl_set:N = \l__scripture_chapter_aboveskip_tl
63 , chapter / aboveskip .value_required:n = true
64 , chapter / aboveskip .initial:n = \c_zero_skip
65 , chapter / align .choices:nn =
66   { left, right }
67   {
68     \tl_if_exist:NF \l__scripture_chapter_align_tl
69     { \tl_new:N \l__scripture_chapter_align_tl }
70     \str_case:Vn \l_keys_choice_str
71     {
72       { left }
73       {
74         \tl_set:Nn \l__scripture_chapter_align_tl
75         { l }
76       }
77       { right }
78       {
79         \tl_set:Nn \l__scripture_chapter_align_tl
80         { r }
81       }
82     }
83   }
84 , chapter / align .value_required:n = true
85 , chapter / align .initial:n = left
86 , chapter / font .tl_set:N = \l__scripture_chap_font_tl
87 , chapter / font .value_required:n = true
88 , chapter / font .initial:n = \bfseries
89 , chapter / colour .tl_set:N = \l__scripture_chapter_colour_tl
90 , chapter / colour .value_required:n = true

```

```

91 , chapter / colour .initial:n = .
92 , chapter / color .meta:n = { chapter / colour = {#1} }
93 , chapter / drop .bool_set:N = \l__scripture_chap_drop_bool
94 , chapter / drop .default:n = true
95 , chapter / drop .initial:n = true
96 , chapter / format .cs_set:Np = \__scripture_chap_format:n #1
97 , chapter / format .value_required:n = true
98 , chapter / format .initial:n = #1
99 , chapter / hide .bool_set_inverse:N = \l__scripture_chap_show_bool
100 , chapter / hide .default:n = true
101 , chapter / language .tl_set:N = \l__scripture_chapter_language_tl
102 , chapter / language .value_required:n = true
103 , chapter / language .initial:n =
104 , chapter / language / variant .tl_set:N =
105     \l__scripture_chapter_language_variant_tl
106 , chapter / language / variant .value_required:n = true
107 , chapter / language / variant .initial:n =
108 , chapter / nodrop .bool_set_inverse:N = \l__scripture_chap_drop_bool
109 , chapter / nodrop .default:n = true
110 , chapter / para .bool_set:N = \l__scripture_chap_para_bool
111 , chapter / para .default:n = true
112 , chapter / para .initial:n = false
113 , chapter / para / aboveskip .tl_set:N =
114     \l__scripture_chap_para_aboveskip_tl
115 , chapter / para / aboveskip .value_required:n = true
116 , chapter / para / aboveskip .initial:n = \bigskipamount
117 , chapter / para / belowskip .tl_set:N =
118     \l__scripture_chap_para_belowskip_tl
119 , chapter / para / belowskip .value_required:n = true
120 , chapter / para / belowskip .initial:n = \medskipamount
121 , chapter / para / indent .code:n =
122     {
123         \bool_set:Nn
124             \l__scripture_chap_para_afterindent_bool
125             { \str_if_eq_p:nn {#1} { true } }
126     }
127 , chapter / para / indent .default:n = true
128 , chapter / para / indent .initial:n = true
129 , chapter / sep .tl_set:N = \l__scripture_chap_sep_tl
130 , chapter / sep .value_required:n = true
131 , chapter / sep .initial:n = 0.5em
132 , chapter / show .bool_set:N = \l__scripture_chap_show_bool
133 , chapter / show .default:n = true
134 , chapter / show .initial:n = true
135 , chapter / showverse .bool_set:N = \l__scripture_chap_show_verse_bool
136 , chapter / showverse .default:n = true
137 , chapter / showverse .initial:n = false
138 , chapter / smash .bool_set:N = \l__scripture_chap_smash_bool
139 , chapter / smash .default:n = true
140 , chapter / smash .initial:n = false
141 , chapter / valign .choices:nn =
142     { bottom, middle, top }
143     {
144         \tl_if_exist:NF \l__scripture_chapter_valign_tl

```

```

145         { \tl_new:N \l__scripture_chapter_valign_tl }
146     \str_case:Vn \l_keys_choice_str
147     {
148         { bottom }
149         {
150             \tl_set:Nn \l__scripture_chapter_valign_tl
151             { b }
152         }
153         { middle }
154         {
155             \tl_set:Nn \l__scripture_chapter_valign_tl
156             { vc }
157         }
158         { top }
159         {
160             \tl_set:Nn \l__scripture_chapter_valign_tl
161             { t }
162         }
163     }
164 }
165 , chapter / valign .value_required:n = true
166 , chapter / valign .initial:n = bottom
167 , chapter / xchar .tl_set:N = \l__scripture_X_char_tl
168 , chapter / xchar .value_required:n = true
169 , chapter / xchar .initial:n = X
170 , defaults .code:n = \cs_if_exist_use:N \__scripture_setup_reset_defaults:
171 , defaults .value_forbidden:n = true
172 , extraskip .tl_set:N = \l__scripture_extraskip_tl
173 , extraskip .value_required:n = true
174 , extraskip .initial:n = \medskipamount
175 , font .tl_set:N = \l__scripture_font_tl
176 , font .value_required:n = true
177 , font .initial:n =
178 , heading / aboveskip .tl_set:N = \l__scripture_heading_aboveskip_tl
179 , heading / aboveskip .value_required:n = true
180 , heading / aboveskip .initial:n = \bigskipamount
181 , heading / afterindent .bool_set:N = \l__scripture_heading_afterindent_bool
182 , heading / afterindent .default:n = true
183 , heading / afterindent .initial:n = false
184 , heading / align .choices:nn =
185     { left, right, center }
186     {
187         \tl_if_exist:NF \l__scripture_heading_align_tl
188         { \tl_new:N \l__scripture_heading_align_tl }
189         \str_case:Vn \l_keys_choice_str
190         {
191             { left }
192             {
193                 \tl_set:Nn \l__scripture_heading_align_tl
194                 { \raggedright }
195             }
196             { right }
197             {
198                 \tl_set:Nn \l__scripture_heading_align_tl

```

```

199         { \raggedleft }
200     }
201     { center }
202     {
203         \tl_set:Nn \l__scripture_heading_align_tl
204         { \centering }
205     }
206 }
207 }
208 , heading / align .value_required:n = true
209 , heading / align .initial:n = left
210 , heading / belowskip .tl_set:N = \l__scripture_heading_belowskip_tl
211 , heading / belowskip .value_required:n = true
212 , heading / belowskip .initial:n = \medskipamount
213 , heading / font .tl_set:N = \l__scripture_heading_font_tl
214 , heading / font .value_required:n = true
215 , heading / font .initial:n = \small\itshape
216 , heading / format .cs_set:Np = \__scripture_heading_format:n #1
217 , heading / format .value_required:n = true
218 , heading / format .initial:n = #1
219 , heading / hide .bool_set_inverse:N = \l__scripture_heading_show_bool
220 , heading / hide .default:n = true
221 , heading / show .bool_set:N = \l__scripture_heading_show_bool
222 , heading / show .default:n = true
223 , heading / show .initial:n = true
224 , indent .bool_set:N = \l__scripture_indent_bool
225 , indent .default:n = true
226 , indent .initial:n = true
227 , inline .bool_set:N = \l__scripture_inline_bool
228 , inline .default:n = true
229 , inline .initial:n = false
230 , inline / begin .tl_set:N = \l__scripture_inline_begin_tl
231 , inline / begin .value_required:n = true
232 , inline / begin .initial:n = ``
233 , inline / end .tl_set:N = \l__scripture_inline_end_tl
234 , inline / end .value_required:n = true
235 , inline / end .initial:n = ''
236 , inline / font .tl_set:N = \l__scripture_inline_font_tl
237 , inline / font .value_required:n = true
238 , inline / font .initial:n =
239 , inline / reference / format .cs_set:Np =
240     \__scripture_inline_ref_format:n #1
241 , inline / reference / format .value_required:n = true
242 , inline / reference / format .initial:n = (#1)
243 , inline / reference / sep .tl_set:N = \l__scripture_inline_ref_sep_tl
244 , inline / reference / sep .value_required:n = true
245 , inline / reference / sep .initial:n = 0.5em
246 , inline / version / delim .tl_set:N = \l__scripture_inline_version_delim_tl
247 , inline / version / delim .value_required:n = true
248 , inline / version / delim .initial:n = \c_space_tl
249 , inline / version / format .cs_set:Np =
250     \__scripture_inline_version_format:n #1
251 , inline / version / format .value_required:n = true
252 , inline / version / format .initial:n = #1

```

```

253 , language .tl_set:N = \l__scripture_main_language_tl
254 , language .value_required:n = true
255 , language .initial:n =
256 , language / variant .tl_set:N = \l__scripture_main_language_variant_tl
257 , language / variant .value_required:n = true
258 , language / variant .initial:n =
259 , leftmargin .tl_set:N = \l__scripture_leftmargin_tl
260 , leftmargin .value_required:n = true
261 , leftmargin .initial:n = \c_zero_dim
262 , mark / verse / format .cs_set:Np = \__scripture_mark_verse_format:nnn
263                                     #1#2#3
264 , mark / verse / format .value_required:n = true
265 , mark / verse / format .initial:n = #1 ~ #2:#3
266 , name / font .tl_set:N = \l__scripture_name_font_tl
267 , name / font .value_required:n = true
268 , name / font .initial:n = \scshape
269 , name / format .cs_set:Np = \__scripture_name_format:n #1
270 , name / format .value_required:n = true
271 , name / format .initial:n = #1
272 , noindent .bool_set_inverse:N = \l__scripture_indent_bool
273 , noindent .default:n = true
274 , parindent .tl_set:N = \l__scripture_parindent_tl
275 , parindent .value_required:n = true
276 , parindent .initial:n = \parindent
277 , parskip .tl_set:N = \l__scripture_parskip_tl
278 , parskip .value_required:n = true
279 , parskip .initial:n = \parskip
280 , pilcrow .bool_set:N = \l__scripture_pilcrow_bool
281 , pilcrow .default:n = true
282 , pilcrow .initial:n = false
283 , pilcrow / sep .tl_set:N = \l__scripture_pilcrow_sep_tl
284 , pilcrow / sep .value_required:n = true
285 , pilcrow / sep .initial:n = 0.25em
286 , redletter .bool_set:N = \l__scripture_red_letter_bool
287 , redletter .default:n = true
288 , redletter .initial:n = false
289 , redletter / colour .tl_set:N = \l__scripture_red_letter_colour_tl
290 , redletter / colour .value_required:n = true
291 , redletter / colour .initial:n = red!80!black
292 , redletter / color .meta:n = { redletter / colour = {#1} }
293 , reference / align .choices:nn =
294     { left, right }
295     {
296         \str_if_exist:NF \l__scripture_ref_align_str
297         { \str_new:N \l__scripture_ref_align_str }
298         \str_set_eq:NN \l__scripture_ref_align_str \l_keys_choice_str
299     }
300 , reference / align .value_required:n = true
301 , reference / align .initial:n = right
302 , reference / colour .tl_set:N = \l__scripture_ref_colour_tl
303 , reference / colour .value_required:n = true
304 , reference / colour .initial:n = .
305 , reference / color .meta:n = { reference / colour = {#1} }
306 , reference / font .tl_set:N = \l__scripture_ref_font_tl

```

```

307 , reference / font .value_required:n = true
308 , reference / font .initial:n = \bfseries
309 , reference / format .cs_set:Np = \__scripture_ref_format:n #1
310 , reference / format .value_required:n = true
311 , reference / format .initial:n = #1
312 , reference / language .tl_set:N = \l__scripture_reference_language_tl
313 , reference / language .value_required:n = true
314 , reference / language .initial:n =
315 , reference / language / variant .tl_set:N =
316 \l__scripture_reference_language_variant_tl
317 , reference / language / variant .value_required:n = true
318 , reference / language / variant .initial:n =
319 , reference / newline .meta:n = { reference / sep = \linewidth }
320 , reference / newline .value_forbidden:n = true
321 , reference / position .choices:nn =
322 { start, end }
323 {
324 \str_if_exist:NF \l__scripture_ref_position_str
325 { \str_new:N \l__scripture_ref_position_str }
326 \str_set_eq:NN \l__scripture_ref_position_str \l_keys_choice_str
327 }
328 , reference / position .value_required:n = true
329 , reference / position .initial:n = end
330 , reference / sep .tl_set:N = \l__scripture_ref_sep_tl
331 , reference / sep .value_required:n = true
332 , reference / sep .initial:n = 2em
333 , reference / start / inline .bool_set_inverse:N =
334 \l__scripture_ref_start_newline_bool
335 , reference / start / inline .default:n = true
336 , reference / start / newline .bool_set:N =
337 \l__scripture_ref_start_newline_bool
338 , reference / start / newline .default:n = true
339 , reference / start / newline .initial:n = false
340 , reference / start / sep .tl_set:N = \l__scripture_ref_start_sep_tl
341 , reference / start / sep .value_required:n = true
342 , reference / start / sep .initial:n = 1em
343 , reference / start / vsep .tl_set:N = \l__scripture_ref_vertical_sep_tl
344 , reference / start / vsep .value_required:n = true
345 , reference / start / vsep .initial:n = 0pt
346 , rightmargin .tl_set:N = \l__scripture_rightmargin_tl
347 , rightmargin .value_required:n = true
348 , rightmargin .initial:n = \c_zero_dim
349 , selah / text .tl_set:N = \l__scripture_selah_text_tl
350 , selah / text .value_required:n = true
351 , selah / text .initial:n = Selah
352 , selah / font .tl_set:N = \l__scripture_selah_font_tl
353 , selah / font .value_required:n = true
354 , selah / font .initial:n = \itshape
355 , selah / format .cs_set:Np = \__scripture_selah_format:n #1
356 , selah / format .value_required:n = true
357 , selah / format .initial:n = #1
358 , selah / sep .tl_set:N = \l__scripture_selah_sep_tl
359 , selah / sep .value_required:n = true
360 , selah / sep .initial:n = 1em

```



```

361 , style .choice:
362 , style / unknown .code:n = \msg_error:nxx
363         { scripture }
364         { unknown-style }
365         { \exp_not:n {#1} }
366 , textright / sep .tl_set:N = \l__scripture_text_right_sep_tl
367 , textright / sep .value_required:n = true
368 , textright / sep .initial:n = 1em
369 , verse .int_gset:N = \g__scripture_verse_int
370 , verse .value_required:n = true
371 , verse .initial:n = \c_one_int
372 , verse / colour .tl_set:N = \l__scripture_verse_colour_tl
373 , verse / colour .value_required:n = true
374 , verse / colour .initial:n = .
375 , verse / color .meta:n = { verse / colour = {#1} }
376 , verse / first .bool_set:N = \l__scripture_verse_first_bool
377 , verse / first .default:n = true
378 , verse / first .initial:n = false
379 , verse / firstformat .cs_set:Np = \__scripture_verse_first_format:n #1
380 , verse / firstformat .value_required:n = true
381 , verse / firstformat .initial:n = #1
382 , verse / firstsep .tl_set:N = \l__scripture_verse_first_sep_tl
383 , verse / firstsep .value_required:n = true
384 , verse / firstsep .initial:n = 0.5em
385 , verse / font .tl_set:N = \l__scripture_verse_font_tl
386 , verse / font .value_required:n = true
387 , verse / font .initial:n =
388 , verse / format .cs_set:Np = \__scripture_verse_format:n #1
389 , verse / format .value_required:n = true
390 , verse / format .initial:n = \textsuperscript{#1}
391 , verse / hide .bool_set_inverse:N = \l__scripture_verse_show_bool
392 , verse / hide .default:n = true
393 , verse / language .tl_set:N = \l__scripture_verse_language_tl
394 , verse / language .value_required:n = true
395 , verse / language .initial:n =
396 , verse / language / variant .tl_set:N =
397     \l__scripture_verse_language_variant_tl
398 , verse / language / variant .value_required:n = true
399 , verse / language / variant .initial:n =
400 , verse / para .bool_set:N = \l__scripture_verse_para_bool
401 , verse / para .default:n = true
402 , verse / para .initial:n = false
403 , verse / para / indent .tl_set:N = \l__scripture_para_indent_tl
404 , verse / para / indent .value_required:n = true
405 , verse / para / indent .initial:n = \l__scripture_parindent_tl
406 , verse / presep .tl_set:N = \l__scripture_verse_presep_tl
407 , verse / presep .value_required:n = true
408 , verse / presep .initial:n = 0em
409 , verse / sep .tl_set:N = \l__scripture_verse_sep_tl
410 , verse / sep .value_required:n = true
411 , verse / sep .initial:n = 0.05em
412 , verse / show .bool_set:N = \l__scripture_verse_show_bool
413 , verse / show .default:n = true
414 , verse / show .initial:n = true

```

```

415 , version .tl_set:N = \l__scripture_version_tl
416 , version .value_required:n = true
417 , version .initial:n =
418 , version / delim .tl_set:N = \l__scripture_version_delim_tl
419 , version / delim .value_required:n = true
420 , version / delim .initial:n = \c_space_tl
421 , version / format .cs_set:Np = \__scripture_version_format:n #1
422 , version / format .value_required:n = true
423 , version / format .initial:n = (#1)
424 , version / language .tl_set:N = \l__scripture_version_language_tl
425 , version / language .value_required:n = true
426 , version / language .initial:n =
427 , version / language / variant .tl_set:N =
428     \l__scripture_version_language_variant_tl
429 , version / language / variant .value_required:n = true
430 , version / language / variant .initial:n =
431 , version / position .choices:nn =
432     { withref, end }
433     {
434         \str_if_exist:NF \l__scripture_version_position_str
435         { \str_new:N \l__scripture_version_position_str }
436         \str_set_eq:NN \l__scripture_version_position_str \l_keys_choice_str
437     }
438 , version / position .value_required:n = true
439 , version / position .initial:n = withref

```

Options for the **center** environment.

```

440 , center / aboveskip .tl_set:N = \l__scripture_center_aboveskip_tl
441 , center / aboveskip .value_required:n = true
442 , center / aboveskip .initial:n = \medskipamount
443 , center / belowskip .tl_set:N = \l__scripture_center_belowskip_tl
444 , center / belowskip .value_required:n = true
445 , center / belowskip .initial:n = \medskipamount
446 , center / leftmargin .tl_set:N = \l__scripture_center_leftmargin_tl
447 , center / leftmargin .value_required:n = true
448 , center / leftmargin .initial:n = \c_zero_dim
449 , center / rightmargin .tl_set:N = \l__scripture_center_rightmargin_tl
450 , center / rightmargin .value_required:n = true
451 , center / rightmargin .initial:n = \c_zero_dim

```

Options for the **flushright** environment.

```

452 , flushright / aboveskip .tl_set:N = \l__scripture_flushright_aboveskip_tl
453 , flushright / aboveskip .value_required:n = true
454 , flushright / aboveskip .initial:n = \medskipamount
455 , flushright / belowskip .tl_set:N = \l__scripture_flushright_belowskip_tl
456 , flushright / belowskip .value_required:n = true
457 , flushright / belowskip .initial:n = \medskipamount
458 , flushright / leftmargin .tl_set:N = \l__scripture_flushright_leftmargin_tl
459 , flushright / leftmargin .value_required:n = true
460 , flushright / leftmargin .initial:n = \c_zero_dim
461 , flushright / rightmargin .tl_set:N =
462     \l__scripture_flushright_rightmargin_tl
463 , flushright / rightmargin .value_required:n = true
464 , flushright / rightmargin .initial:n = \c_zero_dim

```

Options for the `hanging` environment.

```

465 , hanging / aboveskip .tl_set:N = \l__scripture_hanging_aboveskip_tl
466 , hanging / aboveskip .value_required:n = true
467 , hanging / aboveskip .initial:n = \medskipamount
468 , hanging / belowskip .tl_set:N = \l__scripture_hanging_belowskip_tl
469 , hanging / belowskip .value_required:n = true
470 , hanging / belowskip .initial:n = \medskipamount
471 , hanging / chapter / indent .bool_set:N =
472   \l__scripture_hanging_chapter_indent_bool
473 , hanging / chapter / indent .default:n = true
474 , hanging / chapter / indent .initial:n = true
475 , hanging / hang .tl_set:N = \l__scripture_hanging_hang_tl
476 , hanging / hang .value_required:n = true
477 , hanging / hang .initial:n = 2em
478 , hanging / leftmargin .tl_set:N = \l__scripture_hanging_leftmargin_tl
479 , hanging / leftmargin .value_required:n = true
480 , hanging / leftmargin .initial:n = 1em
481 , hanging / rightmargin .tl_set:N = \l__scripture_hanging_rightmargin_tl
482 , hanging / rightmargin .value_required:n = true
483 , hanging / rightmargin .initial:n = \c_zero_dim
484 , hanging / verse / left .bool_set_inverse:N =
485   \l__scripture_hanging_verse_right_bool
486 , hanging / verse / left .default:n = true
487 , hanging / verse / right .bool_set:N =
488   \l__scripture_hanging_verse_right_bool
489 , hanging / verse / right .default:n = true
490 , hanging / verse / right .initial:n = true
491 , hanging / verse / sep .tl_set:N = \l__scripture_hanging_verse_sep_tl
492 , hanging / verse / sep .value_required:n = true
493 , hanging / verse / sep .initial:n = 0.05em

```

Options for the mid-paragraph chapters (including the `midparachap` environment).

```

494 , midparachap / hideverse .bool_set_inverse:N =
495   \l__scripture_midparachap_show_verse_bool
496 , midparachap / hideverse .default:n = true
497 , midparachap / showverse .bool_set:N =
498   \l__scripture_midparachap_show_verse_bool
499 , midparachap / showverse .default:n = true
500 , midparachap / showverse .initial:n = true

```

Options for the `narrow` environment.

```

501 , narrow / aboveskip .tl_set:N = \l__scripture_narrow_aboveskip_tl
502 , narrow / aboveskip .value_required:n = true
503 , narrow / aboveskip .initial:n = \medskipamount
504 , narrow / belowskip .tl_set:N = \l__scripture_narrow_belowskip_tl
505 , narrow / belowskip .value_required:n = true
506 , narrow / belowskip .initial:n = \medskipamount
507 , narrow / leftmargin .tl_set:N = \l__scripture_narrow_leftmargin_tl
508 , narrow / leftmargin .value_required:n = true
509 , narrow / leftmargin .initial:n = 1em
510 , narrow / rightmargin .tl_set:N = \l__scripture_narrow_rightmargin_tl
511 , narrow / rightmargin .value_required:n = true
512 , narrow / rightmargin .initial:n = 1em

```

Options for the `poetry` environment.

```

513 , poetry / aboveskip .tl_set:N = \l__scripture_poetry_aboveskip_tl
514 , poetry / aboveskip .value_required:n = true
515 , poetry / aboveskip .initial:n = \medskipamount
516 , poetry / belowskip .tl_set:N = \l__scripture_poetry_belowskip_tl
517 , poetry / belowskip .value_required:n = true
518 , poetry / belowskip .initial:n = \medskipamount
519 , poetry / bigindent .tl_set:N = \l__scripture_poetry_big_indent_tl
520 , poetry / bigindent .value_required:n = true
521 , poetry / bigindent .initial:n = 4em
522 , poetry / indent .tl_set:N = \l__scripture_poetry_indent_tl
523 , poetry / indent .value_required:n = true
524 , poetry / indent .initial:n = 1em
525 , poetry / leftmargin .tl_set:N = \l__scripture_poetry_leftmargin_tl
526 , poetry / leftmargin .value_required:n = true
527 , poetry / leftmargin .initial:n = 1em
528 , poetry / rightmargin .tl_set:N = \l__scripture_poetry_rightmargin_tl
529 , poetry / rightmargin .value_required:n = true
530 , poetry / rightmargin .initial:n = \c_zero_dim
531 , poetry / verse / left .bool_set_inverse:N =
532   \l__scripture_poetry_verse_right_bool
533 , poetry / verse / left .default:n = true
534 , poetry / verse / right .bool_set:N = \l__scripture_poetry_verse_right_bool
535 , poetry / verse / right .default:n = true
536 , poetry / verse / right .initial:n = true
537 , poetry / verse / sep .tl_set:N = \l__scripture_poetry_verse_sep_tl
538 , poetry / verse / sep .value_required:n = true
539 , poetry / verse / sep .initial:n = 0.05em
540 }

```

Process package options.

```

541 \ProcessKeyOptions
    Create interface to reset all keys to defaults.
542 \tl_new:N \l__scripture_default_options_tl
543 \keys_precompile:nnN { scripture }
544 {
545   , aboveskip                = \c_zero_skip
546   , added / font              =
547   , added / format            = \emph {#1}
548   , after                     =
549   , before                     =
550   , belowskip                 = \c_zero_skip
551   , book                      =
552   , colour                    = .
553   , compact                   = false
554   , chapter                   = \c_one_int
555   , chapter / aboveskip       = \c_zero_skip
556   , chapter / align           = left
557   , chapter / font            = \bfseries
558   , chapter / colour          = .
559   , chapter / drop            = true
560   , chapter / format          = #1
561   , chapter / para            = false
562   , chapter / para / aboveskip = \bigskipamount
563   , chapter / para / belowskip = \medskipamount

```

```

564 , chapter / para / indent      = true
565 , chapter / sep                 = 0.5em
566 , chapter / show               = true
567 , chapter / showverse          = false
568 , chapter / smash              = false
569 , chapter / valign             = bottom
570 , chapter / xchar              = X
571 , extraskip                    = \medskipamount
572 , font                         =
573 , heading / aboveskip           = \bigskipamount
574 , heading / afterindent        = false
575 , heading / align              = left
576 , heading / belowskip          = \medskipamount
577 , heading / font               = \small\itshape
578 , heading / format             = #1
579 , heading / show               = true
580 , indent                       = true
581 , inline                       = false
582 , inline / begin               = ``
583 , inline / end                 = ''
584 , inline / font               =
585 , inline / reference / format  = (#1)
586 , inline / reference / sep     = 0.5em
587 , inline / version / delim    = \c_space_tl
588 , inline / version / format   = #1
589 , language                     =
590 , language / variant           =
591 , leftmargin                   = \c_zero_dim
592 , mark / verse / format       = #1 #2:#3
593 , name / font                 = \scshape
594 , name / format               = #1
595 , parindent                   = \parindent
596 , parskip                     = \parskip
597 , pilcrow                    = false
598 , pilcrow / sep              = 0.25em
599 , redletter                   = false
600 , redletter / colour          = red!80!black
601 , reference / align           = right
602 , reference / colour          = .
603 , reference / font            = \bfseries
604 , reference / format          = #1
605 , reference / language        =
606 , reference / language / variant =
607 , reference / position        = end
608 , reference / sep             = 2em
609 , reference / start / newline = false
610 , reference / start / sep     = 1em
611 , reference / start / vsep    = 0pt
612 , rightmargin                 = \c_zero_dim
613 , salah / text                = Salah
614 , salah / font                = \itshape
615 , salah / format              = #1
616 , salah / sep                 = 1em
617 , textright / sep             = 1em

```

```

618 , verse = \c_one_int
619 , verse / colour = .
620 , verse / first = false
621 , verse / firstformat = #1
622 , verse / firstsep = 0.5em
623 , verse / font =
624 , verse / format = \textsuperscript{#1}
625 , verse / language =
626 , verse / language / variant =
627 , verse / para = false
628 , verse / para / indent = \l__scripture_parindent_tl
629 , verse / presep = 0em
630 , verse / sep = 0.05em
631 , verse / show = true
632 , version =
633 , version / delim = \c_space_tl
634 , version / format = (#1)
635 , version / language =
636 , version / language / variant =
637 , version / position = withref
638 , center / aboveskip = \medskipamount
639 , center / belowskip = \medskipamount
640 , center / leftmargin = \c_zero_dim
641 , center / rightmargin = \c_zero_dim
642 , flushright / aboveskip = \medskipamount
643 , flushright / belowskip = \medskipamount
644 , flushright / leftmargin = \c_zero_dim
645 , flushright / rightmargin = \c_zero_dim
646 , hanging / aboveskip = \medskipamount
647 , hanging / belowskip = \medskipamount
648 , hanging / chapter / indent = true
649 , hanging / hang = 2em
650 , hanging / leftmargin = 1em
651 , hanging / rightmargin = \c_zero_dim
652 , hanging / verse / right = true
653 , hanging / verse / sep = 0.05em
654 , midparachap / showverse = true
655 , narrow / aboveskip = \medskipamount
656 , narrow / belowskip = \medskipamount
657 , narrow / leftmargin = 1em
658 , narrow / rightmargin = 1em
659 , poetry / aboveskip = \medskipamount
660 , poetry / belowskip = \medskipamount
661 , poetry / bigindent = 4em
662 , poetry / indent = 1em
663 , poetry / leftmargin = 1em
664 , poetry / rightmargin = \c_zero_dim
665 , poetry / verse / right = true
666 , poetry / verse / sep = 0.05em
667 }
668 \l__scripture_default_options_tl
669 \cs_new_protected:Nn \__scripture_setup_reset_defaults:
670 {
671   \l__scripture_default_options_tl

```

672 }

`_scripture_setup_style:nnn` Specify options which can later be activated using the **style** option. If the first argument is **true**, then the options are appended rather than replaced.

```
673 \cs_new_protected:Nn \_scripture_setup_style:nnn
674 {
675   \bool_if:NTF #1
676   {
677     \tl_if_exist:cF { g__scripture_style_ #2 _tl }
678     { \tl_new:c { g__scripture_style_ #2 _tl } }
679     \_scripture_setup_style:nne { \c_false_bool } {#2}
680     { \tl_use:c { g__scripture_style_ #2 _tl } , #3 }
681   }
682   {
683     \tl_gset:cn { g__scripture_style_ #2 _tl } {#3}
684     \keys_define:nn { scripture }
685     { style / #2 .meta:n = {#3} }
686   }
687 }
688 \cs_generate_variant:Nn \_scripture_setup_style:nnn { nne }
```

(End of definition for `_scripture_setup_style:nnn`.)

3.3 Hooks

scripture/chap/before Hooks for before and after chapters and verses. The current chapter and verse is available in the hooks as `\scripturecurrentchapter` and `\scripturecurrentverse` respectively. These are global.

```
689 \hook_new_pair:nn { scripture / chap / before } { scripture / chap / after }
690 \hook_new_pair:nn { scripture / verse / before } { scripture / verse / after }
```

(End of definition for `scripture/chap/before` and others. These variables are documented on page 12.)

scripture/pilcrow These are internal hooks used to insert a pilcrow at the beginning of a paragraph. Pilcrow are activated using the **pilcrow** option and are only in effect when the **verse/para** and/or **compact** options are in use.

```
691 \hook_new:n { scripture / pilcrow }
692 \hook_new:n { scripture / poetry / pilcrow }
```

(End of definition for `scripture/pilcrow` and `scripture/poetry/pilcrow`. These variables are documented on page ??.)

3.4 Marks

scripture/heading Independent marks are set up for headings and the current reference. These are called for each `\heading` and `\vs` respectively. The **scripture/verse** mark is only set when a Bible book name has been set with the **book** option.

```
693 \mark_new_class:n { scripture / heading }
694 \mark_new_class:n { scripture / verse }
```

(End of definition for `scripture/heading` and `scripture/verse`. These variables are documented on page 13.)

`_scripture_insert_verse_mark:`

```

695 \cs_new_protected:Nn \__scripture_insert_verse_mark:
696 {
697   \tl_if_empty:NF \scripturecurrentbook
698   {
699     \mark_insert:nn { scripture / verse }
700     {
701       \__scripture_mark_verse_format:nnn
702       { \scripturecurrentbook }
703       { \scripturecurrentchapter }
704       { \scripturecurrentverse }
705     }
706   }
707 }

```

(End of definition for __scripture_insert_verse_mark:.)

And the definition of `\name`, `\GOD` and `\LORD` to the insertmark hook.

```

708 \hook_gput_code:nnn { insertmark } { scripture }
709 {
710   \cs_set_eq:NN \name \__scripture_name:n
711   \cs_set_nopar:Npn \LORD { \__scripture_name:n { Lord } }
712   \cs_set_nopar:Npn \GOD { \__scripture_name:n { God } }
713 }

```

3.5 The main **scripture** environment

`\l__scripture_active_bool` Flags to ensure that environments are not nested.

```

\l__scripture_active_inner_bool
714 \bool_new:N \l__scripture_active_bool
715 \bool_new:N \l__scripture_active_inner_bool

```

(End of definition for \l__scripture_active_bool and \l__scripture_active_inner_bool.)

`\g__scripture_first_verse_set_bool` Set to **true** when the first `\ch` or `\vs` has been set. This is used to ensure automatic chapter and verse numbering starts at the right place.

```

716 \bool_new:N \g__scripture_first_verse_set_bool

```

(End of definition for \g__scripture_first_verse_set_bool.)

`\g__scripture_mode_vertical_inner_bool` Whether an inner environment is starting in vertical mode. This flag is only used with the **compact** and **verse/para** options.

```

717 \bool_new:N \g__scripture_mode_vertical_inner_bool

```

(End of definition for \g__scripture_mode_vertical_inner_bool.)

`\l__scripture_clubpenalty_saved_int` Holds saved club penalty so we can change and restore it around chapter breaks.

```

718 \int_new:N \l__scripture_clubpenalty_saved_int

```

(End of definition for \l__scripture_clubpenalty_saved_int.)

`\g__scripture_prev_inner_below_skip` Save the last below skip of an inner environment so it can be removed if the main scripture environment ends with an inner environment.

```

719 \skip_new:N \g__scripture_prev_inner_below_skip

```

(End of definition for \g__scripture_prev_inner_below_skip.)

`\l__scripture_outer_itemindent_dim` Holds the `\itemindent` of the outer list. Used to calculate the `\itemindent` to retain correct indenting when a list environment is the first thing in an outer list.

720 `\dim_new:N \l__scripture_outer_itemindent_dim`

(End of definition for `\l__scripture_outer_itemindent_dim`.)

`__scripture_setup_list_noindent:n` Set `\noindent` to do a negative horizontal skip a list is the first item in a list then restore it at the end of the paragraph. This allows `\itemindent` to be set to a negative value and still have `\noindent` behave as expected.

```

721 \cs_new_protected_nopar:Nn \__scripture_setup_list_noindent:n
722 {
723   \legacy_if:nT { @newlist }
724   {
725     \cs_set_protected_nopar:Npn \noindent
726     {
727       \skip_horizontal:n { -#1 }
728       \bool_if:NT \l__scripture_verse_para_bool
729       {
730         \peek_analysis_map_inline:n
731         {
732           \bool_lazy_or:nnTF
733             { \exp_args:No \token_if_eq_meaning_p:NN {####1} \vs }
734             { \exp_args:No \token_if_eq_meaning_p:NN {####1} \newvs }
735           {
736             \bool_gset_true:N
737               \g__scripture_suppress_next_verse_para_bool
738               \__scripture_set_pilcrow_hook:
739             }
740             {
741               \str_if_eq:VnF \l__scripture_currenvir_str { poetry }
742               { \__scripture_pilcrow_output: }
743             }
744             \peek_analysis_map_break:n {####1}
745           }
746         }
747       }
748       \hook_gput_next_code:nn { para / end }
749       {
750         \bool_if:NTF \l__scripture_verse_para_bool
751         {
752           \str_if_eq:VnTF \l__scripture_currenvir_str { poetry }
753           { \cs_set_eq:NN \noindent \__scripture_noindent_saved: }
754           { \cs_set_eq:NN \noindent \__scripture_para_noindent: }
755         }
756         { \cs_set_eq:NN \noindent \__scripture_noindent_saved: }
757       }
758     }
759   }

```

(End of definition for `__scripture_setup_list_noindent:n`.)

`__scripture_skip_diff:NN` Find the difference between two skips while avoiding negative stretch and shrink components.

760 `\cs_new_protected:Nn __scripture_skip_diff:NN`

```

761 {
762   \skip_eval:n
763   {
764     \dim_eval:n { \dimexpr #1 - \dimexpr #2 }
765     ~ plus ~
766     \dim_max:nn { \gluestretch #1 - \gluestretch #2 } \c_zero_dim
767     ~ minus ~
768     \dim_max:nn { \glueshrink #1 - \glueshrink #2 } \c_zero_dim
769   }
770 }

```

(End of definition for `__scripture_skip_diff:NN`.)

`__scripture_extra_skip:` Insert a little extra vertical space between two paragraphs. This function tests if the previous paragraph is only one line in length and began with a `\ch` function. If this condition is met, more space is added if necessary to avoid the following paragraph clashing with the drop chapter.

```

771 \cs_new_protected:Nn \__scripture_extra_skip:
772 {
773   \mode_if_vertical:F
774   {
775     \bool_if:NTF \l__scripture_verse_para_bool
776     { \para_end: }
777     { \par }
778   }
779   \int_compare:nNnT
780   { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
781   {
782     \dim_compare:nNnT
783     { \parskip + \glueexpr \l__scripture_extraskip_tl }
784     <
785     { \baselineskip }
786     {
787       \addvspace { \baselineskip }
788     }
789     \int_gzero:N \g__scripture_chap_par_prevgraf_int
790   }
791   \addvspace { \l__scripture_extraskip_tl }
792 }

```

(End of definition for `__scripture_extra_skip:.`)

`__scripture_text_right:nn` Align a word to the right and finish a paragraph, pushing to the next line if there is not space for the word.

```

793 \cs_new_protected:Nn \__scripture_text_right:nn
794 {
795   \bool_lazy_or:nnTF
796   { \str_if_eq_p:Vn \l__scripture_currenvir_str { flushright } }
797   { \str_if_eq_p:Vn \l__scripture_currenvir_str { center } }
798   {
799     \unskip
800     \break
801     \strut
802     \hfill

```

```

803         #2
804         \para_end:
805     }
806     {
807     {
808         \unskip
809         \hfil
810         \penalty 50
811         \skip_horizontal:n {#1}
812         \hbox:n {}
813         \nobreak
814         \hfill
815         #2
816         \skip_zero:N \parfillskip
817         \int_zero:N \finalhyphendemerits
818         \para_end:
819     }
820     }
821 }

```

(End of definition for _scripture_text_right:nn.)

3.5.1 Formatting and printing quotation references

\l_scripture_ref_box Box to hold the formatted reference.

```

822 \box_new:N \l\_scripture_ref_box

```

(End of definition for \l_scripture_ref_box.)

_scripture_format_full_ref:nn Format a scripture quotation with its reference.

```

823 \cs_new_protected:Nn \_scripture_format_full_ref:nn
824 {
825     \group_begin:
826     \l\_scripture_ref_font_tl
827     \str_if_eq:VnF \l\_scripture_ref_colour_tl { . }
828     { \exp_args:NV \color_select:n \l\_scripture_ref_colour_tl }
829     \_scripture_select_language:n { reference }
830     \_scripture_ref_format:n
831     {
832         #1

```

Append a Bible version if it has been specified.

```

833         \tl_if_empty:oF {#2}
834         {
835             \tl_if_empty:oF {#1}
836             { \l\_scripture_version_delim_tl }
837             \_scripture_select_language:n { version }
838             \_scripture_version_format:n {#2}
839         }
840     }
841     \group_end:
842 }

```

(End of definition for _scripture_format_full_ref:nn.)

`__scripture_reference:n` Output the reference for the Scripture quotation.

```

843 \cs_new_protected:Nn \__scripture_reference:n
844 {
845   \hbox_set:Nn \l__scripture_ref_box
846   {
847     \__scripture_format_full_ref:nn {#1} \l__scripture_version_tl
848   }
849   \dim_compare:nNnT
850   {
851     \box_wd:N \l__scripture_ref_box +
852     \g__scripture_final_line_dim + \l__scripture_ref_sep_tl
853   }
854   >
855   { \linewidth }
856   {
857     \skip_vertical:N \baselineskip
858     \str_if_eq:VnTF \l__scripture_ref_align_str { left }
859     {
860       \int_compare:nNnT
861       { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
862       { \skip_vertical:N \baselineskip }
863     }
864   }
865   \str_if_eq:VnTF \l__scripture_ref_align_str { right }
866   { \hfill }
867   {
868     \dim_compare:nNnT
869     {
870       \box_wd:N \l__scripture_ref_box +
871       \g__scripture_final_line_dim + \l__scripture_ref_sep_tl
872     }
873     <
874     { \linewidth }
875     {
876       \skip_horizontal:n
877       { \g__scripture_final_line_dim + \l__scripture_ref_sep_tl }
878     }
879   }
880   \box_use:N \l__scripture_ref_box
881 }

```

(End of definition for __scripture_reference:n.)

`__scripture_reference_start_inline:` Output the reference for the Scripture quotation when inline at the start.

```

882 \cs_new_protected:Nn \__scripture_reference_start_inline:
883 {
884   \tl_if_empty:NF \l__scripture_ref_tl
885   {
886     \str_if_eq:VnTF \l__scripture_version_position_str { withref }
887     {
888       \__scripture_format_full_ref:nn
889       { \l__scripture_ref_tl } { \l__scripture_version_tl }
890     }
891     { \__scripture_format_full_ref:nn { \l__scripture_ref_tl } { } }

```

```

892     \skip_horizontal:N \l__scripture_ref_start_sep_tl
893   }
894 }

```

(End of definition for `__scripture_reference_start_inline:`)

`__scripture_reference_start_newline:` Output the reference for the Scripture quotation when on a new line at the start.

```

895 \cs_new_protected:Nn \__scripture_reference_start_newline:
896 {
897   \tl_if_empty:NF \l__scripture_ref_tl
898   {
899     \bool_gset_eq:NN \g_tmpa_bool \g__scripture_suppress_next_pilcrow_bool
900     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
901     \noindent
902     \bool_gset_eq:NN \g__scripture_suppress_next_pilcrow_bool \g_tmpa_bool
903     \str_if_eq:VnF \l__scripture_currenvir_str { scripture }
904     { \skip_horizontal:n { - \leftmargin } }
905     \str_if_eq:VnTF \l__scripture_version_position_str { withref }
906     {
907       \__scripture_format_full_ref:nn
908       { \l__scripture_ref_tl } { \l__scripture_version_tl }
909     }
910     { \__scripture_format_full_ref:nn { \l__scripture_ref_tl } { } }
911     \bool_lazy_or:nnT
912     { \str_if_eq_p:Vn \l__scripture_currenvir_str { flushright } }
913     { \str_if_eq_p:Vn \l__scripture_currenvir_str { center } }
914     {
915       \hfill
916       \strut
917     }
918     \bool_if:NTF \l__scripture_compact_bool
919     { \para_end: }
920     {
921       \bool_lazy_all:nTF
922       {
923         { \l__scripture_verse_para_bool }
924         {
925           \bool_not_p:n
926           { \str_if_eq_p:Vn \l__scripture_currenvir_str { hanging } }
927         }
928         {
929           \bool_not_p:n
930           { \str_if_eq_p:Vn \l__scripture_currenvir_str { poetry } }
931         }
932       }
933       { \para_end: }
934       { \par }
935     }
936     \nobreak
937     \@afterheading
938     \addvspace { \l__scripture_ref_vertical_sep_tl }
939   }
940 }

```

(End of definition for `__scripture_reference_start_newline:`)

_scripture_reference_start_peek: Peek ahead at the beginning of the **scripture** environment to control how references are inserted at the start of a quotation.

```

941 \cs_new_protected:Nn \__scripture_reference_start_peek:
942 {
943   \peek_remove_spaces:n
944   {
945     \bool_lazy_and:nnTF
946     { \str_if_eq_p:Vn \l__scripture_ref_position_str { start } }
947     { \bool_not_p:n { \tl_if_empty_p:N \l__scripture_ref_tl } }
948     {
949       \bool_lazy_or:nnF
950       { \str_if_eq_p:Vn \l__scripture_currenvir_str { scripture } }
951       { \str_if_eq_p:Vn \l__scripture_currenvir_str { midparachap } }
952       {
953         \bool_if:NF \l__scripture_compact_bool
954         { \bool_set_true:N \l__scripture_ref_start_newline_bool }
955       }
956     }
957     \peek_analysis_map_inline:n
958     {
959       \bool_case:nF
960       {
961         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \heading }
962         { \__scripture_reference_start_newline: }
963         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \noindent }
964         {
965           \bool_if:NTF \l__scripture_ref_start_newline_bool
966           { \__scripture_reference_start_newline: }
967           { \__scripture_reference_start_inline: }
968         }
969       }
970       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nohang }
971       {
972         \bool_if:NTF \l__scripture_ref_start_newline_bool
973         { \__scripture_reference_start_newline: }
974         { \__scripture_reference_start_inline: }
975       }
976       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nopilcrow }
977       {
978         \bool_if:NTF \l__scripture_ref_start_newline_bool
979         { \__scripture_reference_start_newline: }
980         { \__scripture_reference_start_inline: }
981         \peek_analysis_map_break:n
982         {
983           ##1
984           \__scripture_fix_protrusion:n { }
985         }
986       }
987     }
988     { \exp_args:No \token_if_eq_meaning_p:NN {##1} \begin }
989     {
990       \bool_if:NF \l__scripture_indent_bool
991       {
992         \hook_gput_next_code:nn { para / begin }
993         {
994           \bool_lazy_or:nnT
995           {

```

```

993         \str_if_eq_p:Vn
994         \l__scripture_currenvir_str
995         { midparachap }
996     }
997     {
998         \str_if_eq_p:Vn
999         \l__scripture_currenvir_str
1000         { scripture }
1001     }
1002     { \para_omit_indent: }
1003 }
1004 }
1005 }
1006 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \ch }
1007 {
1008     \bool_if:NTF \l__scripture_ref_start_newline_bool
1009     { \__scripture_reference_start_newline: }
1010     {
1011         \cs_set_eq:NN \__scripture_temp_ch \ch
1012         \RenewDocumentCommand \ch { s o m }
1013         {
1014             \tl_if_novalue:nF {##2}
1015             {
1016                 \tl_set:Nn \l__scripture_X_char_tl {##2}
1017             }
1018             \cs_set_eq:NN \ch \__scripture_temp_ch:
1019         }
1020         \__scripture_reference_start_inline:
1021     }
1022 }
1023 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newch }
1024 {
1025     \bool_if:NTF \l__scripture_ref_start_newline_bool
1026     { \__scripture_reference_start_newline: }
1027     {
1028         \cs_set_eq:NN \__scripture_temp_newch: \newch
1029         \RenewDocumentCommand \newch { s o }
1030         {
1031             \tl_if_novalue:nF {##2}
1032             {
1033                 \tl_set:Nn \l__scripture_X_char_tl {##2}
1034             }
1035             \cs_set_eq:NN \newch \__scripture_temp_newch:
1036         }
1037         \__scripture_reference_start_inline:
1038     }
1039 }
1040 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1041 {
1042     \bool_if:NTF \l__scripture_ref_start_newline_bool
1043     { \__scripture_reference_start_newline: }
1044     {
1045         \cs_set_eq:NN \__scripture_temp_vs: \vs
1046         \RenewDocumentCommand \vs { m }

```

```

1047         { \cs_set_eq:NN \vs \__scripture_temp_vs: }
1048         \__scripture_reference_start_inline:
1049     }
1050 }
1051 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1052 {
1053     \bool_if:NTF \l__scripture_ref_start_newline_bool
1054     { \__scripture_reference_start_newline: }
1055     {
1056         \cs_set_eq:NN \__scripture_temp_newvs: \newvs
1057         \RenewDocumentCommand \newvs { }
1058         { \cs_set_eq:NN \newvs \__scripture_temp_newvs: }
1059         \__scripture_reference_start_inline:
1060     }
1061 }
1062 }
1063 {
1064     \bool_if:NTF \l__scripture_ref_start_newline_bool
1065     {
1066         \__scripture_reference_start_newline:
1067         \bool_if:NTF \l__scripture_compact_bool
1068         {
1069             \bool_if:NT \g__scripture_mode_vertical_inner_bool
1070             { \__scripture_pilcrow_output: }
1071         }
1072         {
1073             \str_if_eq:VnF
1074             \l__scripture_currenvir_str { poetry }
1075             { \__scripture_pilcrow_output: }
1076         }
1077     }
1078     { \__scripture_reference_start_inline: }
1079 }
1080 \exp_args:No \token_if_eq_meaning:NNF {##1} \begin
1081 {
1082     \tl_clear:N \l__scripture_ref_tl
1083     \str_if_eq:VnF \l__scripture_version_position_str { end }
1084     { \tl_clear:N \l__scripture_version_tl }
1085 }
1086 \peek_analysis_map_break:n
1087 { \__scripture_fix_protrusion:o {##1} }
1088 }
1089 }
1090 {
1091     \peek_analysis_map_inline:n
1092     {
1093         \bool_case:nF
1094         {
1095             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \heading }
1096             { }
1097             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \noindent }
1098             { }
1099             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nohang }
1100             { }

```



```

1101 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nopilcrow }
1102 {
1103   \peek_analysis_map_break:n
1104   {
1105     ##1
1106     \__scripture_fix_protrusion:n { }
1107   }
1108 }
1109 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \begin }
1110 {
1111   \bool_if:NF \l__scripture_indent_bool
1112   {
1113     \hook_gput_next_code:nn { para / begin }
1114     {
1115       \bool_lazy_or:nnT
1116       {
1117         \str_if_eq_p:Vn
1118         \l__scripture_currenvir_str
1119         { midparachap }
1120       }
1121       {
1122         \str_if_eq_p:Vn
1123         \l__scripture_currenvir_str
1124         { scripture }
1125       }
1126       { \para_omit_indent: }
1127     }
1128   }
1129 }
1130 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \ch }
1131 { }
1132 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newch }
1133 { }
1134 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1135 {
1136   \mode_if_vertical:F
1137   { \__scripture_set_pilcrow_hook: }
1138 }
1139 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \news }
1140 {
1141   \mode_if_vertical:F
1142   { \__scripture_set_pilcrow_hook: }
1143 }
1144 }
1145 {
1146   \bool_if:NTF \l__scripture_compact_bool
1147   {
1148     \bool_if:NT \g__scripture_mode_vertical_inner_bool
1149     { \__scripture_pilcrow_output: }
1150   }
1151   {
1152     \str_if_eq:VnF \l__scripture_currenvir_str { poetry }
1153     { \__scripture_pilcrow_output: }
1154   }

```

```

1155         }
1156         \peek_analysis_map_break:n
1157         {
1158             \__scripture_fix_protrusion:o {##1}
1159         }
1160     }
1161 }
1162 }
1163 }

```

(End of definition for __scripture_reference_start_peek:.)

__scripture_compact_par: \par macro used when the **compact** option is in use.

```

1164 \cs_new_protected:Nn \__scripture_compact_par:
1165 {
1166     \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
1167     \dim_set:Nn \parindent { \l__scripture_parindent_tl }
1168     \peek_remove_spaces:n
1169     {
1170         \peek_analysis_map_inline:n
1171         {
1172             \bool_case:n
1173             {
1174                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \begin }
1175                 { \bool_gset_true:N \g__scripture_mode_vertical_inner_bool }
1176                 { \str_if_eq_p:Vn \l__scripture_currenvir_str { poetry } }
1177                 { }
1178                 { \exp_args:No \token_if_letter_p:N {##1} }
1179                 { \__scripture_pilcrow_output: }
1180                 { \exp_args:No \token_if_active_p:N {##1} }
1181                 { \__scripture_pilcrow_output: }
1182                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1183                 { \__scripture_set_pilcrow_hook: }
1184                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1185                 { \__scripture_set_pilcrow_hook: }
1186             }
1187             \peek_analysis_map_break:n {##1}
1188         }
1189     }
1190 }

```

(End of definition for __scripture_compact_par:.)

__scripture_para_noindent: \noindent macro used when the **verse/para** option is in use.

```

1191 \cs_new_protected:Nn \__scripture_para_noindent:
1192 {
1193     \legacy_if:nTF { @newlist }
1194     {
1195         \__scripture_noindent_saved:
1196         \peek_analysis_map_inline:n
1197         {
1198             \bool_lazy_or:nnTF
1199             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1200             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1201             { \bool_gset_true:N \g__scripture_suppress_next_verse_para_bool }

```

```

1202         { \_scripture_pilcrow_output: }
1203         \peek_analysis_map_break:n {##1}
1204     }
1205 }
1206 {
1207     \peek_analysis_map_inline:n
1208     {
1209         \bool_lazy_or:nnTF
1210         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1211         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1212         {
1213             \bool_if:NT \l__scripture_verse_para_bool
1214             { \para_end: }
1215             \_scripture_noindent_saved:
1216             \bool_gset_true:N \g__scripture_suppress_next_verse_para_bool
1217             \_scripture_set_pilcrow_hook:
1218         }
1219         { \_scripture_pilcrow_output: }
1220         \peek_analysis_map_break:n {##1}
1221     }
1222 }
1223 }

```

(End of definition for _scripture_para_noindent:.)

\g__scripture_para_mode_vertical_bool Flag that _scripture_para_par: has been encountered and pilcrow not yet printed.

```

1224 \bool_new:N \g__scripture_para_mode_vertical_bool

```

(End of definition for \g__scripture_para_mode_vertical_bool.)

\g__scripture_post_env_indent_box Box to store the indent after an inner environment in the case where it is dropped by _scripture_dopeend:. This box is then inserted if the next thing is a \vs. Only used when the **verse/para** option is active.

```

1225 \box_new:N \g__scripture_post_env_indent_box

```

(End of definition for \g__scripture_post_env_indent_box.)

_scripture_para_par: \par macro used when the **verse/para** option is in use.

```

1226 \cs_new_protected:Nn \_scripture_para_par:
1227 {
1228     \bool_gset_true:N \g__scripture_para_mode_vertical_bool
1229     \dim_set:Nn \parindent { \l__scripture_parindent_tl }
1230     \peek_remove_spaces:n
1231     {
1232         \peek_analysis_map_inline:n
1233         {
1234             \bool_case:n
1235             {
1236                 { \exp_args:No \token_if_letter_p:N {##1} }
1237                 { \_scripture_pilcrow_output: }
1238                 { \exp_args:No \token_if_active_p:N {##1} }
1239                 { \_scripture_pilcrow_output: }
1240                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1241                 { \_scripture_set_pilcrow_hook: }
1242                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }

```

```

1243         { \_scripture_set_pilcrow_hook: }
1244     }
1245     \peek_analysis_map_break:n {##1}
1246 }
1247 }
1248 }

```

(End of definition for _scripture_para_par:.)

_scripture_doendpe: Custom version of \@doendpe macro used when the **verse/para** option is in use. This handles pilcrow after inner environments.

```

1249 \cs_new_protected:Nn \_scripture_doendpe:
1250 {
1251     \box_gclear:N \g__scripture_post_env_indent_box
1252     \legacy_if_set_true:n { @endpe }
1253     \cs_set_protected:Npn \par
1254     {
1255         \@restorepar
1256         \clubpenalty \@clubpenalty
1257         \everypar { }
1258         \par
1259         \legacy_if_set_false:n { @endpe }
1260         \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
1261         \_scripture_para_par:
1262     }
1263     \everypar
1264     {
1265         \box_gset_to_last:N \g__scripture_post_env_indent_box
1266         \hook_gput_next_code:nm { para / after }
1267         { \box_gclear:N \g__scripture_post_env_indent_box }
1268         \everypar { }
1269         \legacy_if_set_false:n { @endpe }
1270         \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1271         \bool_if:NT \l__scripture_verse_para_bool
1272         {
1273             \box_set_wd:Nn
1274             \g__scripture_post_env_indent_box
1275             { \l__scripture_para_indent_tl }
1276         }
1277     }
1278 }

```

(End of definition for _scripture_doendpe:.)

_scripture_fix_protrusion:n The use of \peek macros breaks protrusion at the beginning of list environments. This function checks conditions and add \leftprotrusion.

_scripture_fix_protrusion:o

```

1279 \cs_new_protected:Nn \_scripture_fix_protrusion:n
1280 {
1281     \bool_if:NTF \l__scripture_compact_bool
1282     {
1283         \str_if_eq:VnT \l__scripture_currenvir_str { scripture }
1284         { \cs_if_exist_use:N \leftprotrusion }
1285     }
1286     { \cs_if_exist_use:N \leftprotrusion }

```

```

1287     #1
1288   }
1289   \cs_generate_variant:Nn \__scripture_fix_protrusion:n { o }

```

(End of definition for __scripture_fix_protrusion:n.)

3.5.2 Red lettering

`\g_scripture_red_letter_active_bool` Flag indicating if currently in a red letter block of text.

```

1290 \bool_new:N \g_scripture_red_letter_active_bool

```

(End of definition for \g_scripture_red_letter_active_bool.)

`__scripture_red_letter_on:` Mark the beginning of a section of text attributed to Jesus Christ.

```

1291 \cs_new_protected:Nn \__scripture_red_letter_on:
1292 {
1293   \bool_if:NT \l__scripture_red_letter_bool
1294   {
1295     \bool_gset_true:N \g_scripture_red_letter_active_bool
1296     \exp_args:NV \color_select:n \l__scripture_red_letter_colour_tl
1297     \str_if_eq:VnF \l__scripture_currenvir_str { scripture }
1298     {
1299       \hook_gput_next_code:nn
1300       { env / \l__scripture_currenvir_str / after }
1301       {
1302         \exp_args:NV \color_select:n \l__scripture_red_letter_colour_tl
1303       }
1304     }
1305   }
1306 }

```

(End of definition for __scripture_red_letter_on:.)

`__scripture_red_letter_off:` Mark the beginning of a section of text attributed to Jesus Christ.

```

1307 \cs_new_protected:Nn \__scripture_red_letter_off:
1308 {
1309   \bool_if:NT \l__scripture_red_letter_bool
1310   {
1311     \color_select:n { scripture default colour }
1312     \bool_gset_false:N \g_scripture_red_letter_active_bool
1313     \str_if_eq:VnF \l__scripture_currenvir_str { scripture }
1314     {
1315       \hook_gput_next_code:nn
1316       { env / \l__scripture_currenvir_str / after }
1317       {
1318         \color_select:n { scripture default colour }
1319       }
1320     }
1321   }
1322 }

```

(End of definition for __scripture_red_letter_off:.)

3.5.3 Selecting language

`_scripture_select_language:n` Select specified language if required. #1 is one of main, chapter, reference, verse or version.

```

1323 \cs_new_protected:Nn \_scripture_select_language:n
1324 {
1325   \tl_set_eq:Nc \l__scripture_language_tl
1326   { l__scripture_#1_language_tl }
1327   \tl_set_eq:Nc \l__scripture_language_variant_tl
1328   { l__scripture_#1_language_variant_tl }
1329   \tl_if_empty:NF \l__scripture_language_tl
1330   {
1331     \@ifpackageloaded { babel }
1332     {
1333       \selectlanguage { \l__scripture_language_tl }
1334     }
1335     {
1336       \@ifpackageloaded { polyglossia }
1337       {
1338         \iflanguageloaded { \l__scripture_language_tl }
1339         {
1340           \tl_if_empty:NTF \l__scripture_language_variant_tl
1341           {
1342             \selectlanguage { \l__scripture_language_tl }
1343           }
1344           {
1345             \selectlanguage
1346             [ variant = \l__scripture_language_variant_tl ]
1347             { \l__scripture_language_tl }
1348           }
1349         }
1350         {
1351           \msg_error:nnx
1352           { scripture }
1353           { language-error }
1354           { \l__scripture_language_tl }
1355         }
1356       }
1357       {
1358         \msg_warning:nnx
1359         { scripture }
1360         { language-warning }
1361         { \l__scripture_language_tl }
1362       }
1363     }
1364   }
1365 }
```

(End of definition for _scripture_select_language:n.)

3.5.4 Formatting and printing chapter numbers

`\l__scripture_chap_box` Boxes use to store and calculate drop chapter.
`l__scripture_chap_tmp_box` `\box_new:N \l__scripture_chap_box`

1367 `\box_new:N \l__scripture_chap_tmp_box`

(End of definition for \l__scripture_chap_box and l__scripture_chap_tmp_box.)

`\l_scripture_chap_fontsize_dim` Calculated font size of a drop chapter number.

1368 `\dim_new:N \l__scripture_chap_fontsize_dim`

(End of definition for \l__scripture_chap_fontsize_dim.)

`\l_scripture_chap_height_dim` Calculated height of a drop chapter number.

1369 `\dim_new:N \l__scripture_chap_height_dim`

(End of definition for \l__scripture_chap_height_dim.)

`\g__scripture_chap_width_dim` Calculated width of drop chapter number.

1370 `\dim_new:N \g__scripture_chap_width_dim`

(End of definition for \g__scripture_chap_width_dim.)

`\g_scripture_chap_par_prevgraf_int` Store the number of lines a chapter paragraph takes up.

1371 `\int_new:N \g__scripture_chap_par_prevgraf_int`

(End of definition for \g__scripture_chap_par_prevgraf_int.)

`\l_scripture_chap_X_height_dim` Calculated height of X at current font size. This is used as the height for a drop chapter.

1372 `\dim_new:N \l__scripture_chap_X_height_dim`

(End of definition for \l__scripture_chap_X_height_dim.)

`\l_scripture_para_chap_indent_dim` Indent used to center chapters when placed in their own paragraph.

1373 `\dim_new:N \l__scripture_para_chap_indent_dim`

(End of definition for \l__scripture_para_chap_indent_dim.)

`\l_scripture_nodrop_chap_coffin` Coffin to place non-drop chapters inside.

1374 `\coffin_new:N \l__scripture_nodrop_chap_coffin`

(End of definition for \l__scripture_nodrop_chap_coffin.)

`\l_scripture_nodrop_chap_voffset_dim` Vertical offset to use when setting non-drop chapters.

1375 `\dim_new:N \l__scripture_nodrop_chap_voffset_dim`

(End of definition for \l__scripture_nodrop_chap_voffset_dim.)

`__scripture_get_X_height:` Calculate the height of `\l__scripture_X_char_tl` for vertical alignment of non-drop chapters.

1376 `\cs_new_protected:Nn __scripture_get_X_height:`

1377 `{`

1378 `\hbox_set:Nn \l__scripture_chap_tmp_box { \l__scripture_X_char_tl }`

1379 `\dim_set:Nn \l__scripture_chap_X_height_dim`

1380 `{ \box_ht:N \l__scripture_chap_tmp_box }`

1381 `}`

(End of definition for __scripture_get_X_height:.)

`_scripture_drop_chap_set_up:n` Set up chapter drop cap and calculate its various dimensions.

```

1382 \cs_new_protected:Nn \__scripture_drop_chap_set_up:n
1383 {
1384   \tl_gset:Nn \scripturecurrentchapter {#1}
1385   \tl_gset:Nn \scripturecurrentverse { 1 }
1386   \__scripture_get_X_height:
1387   \dim_set:Nn \l__scripture_chap_height_dim
1388     { \baselineskip + \l__scripture_chap_X_height_dim }
1389   \dim_set:Nn \l__scripture_chap_fontsize_dim
1390     {
1391       \f@size pt *
1392       \dim_ratio:nn
1393         { \l__scripture_chap_height_dim }
1394         { \l__scripture_chap_X_height_dim }
1395     }
1396   \hbox_set:Nn \l__scripture_chap_tmp_box
1397     {
1398       \l__scripture_chap_font_tl
1399       \fontsize
1400         { \l__scripture_chap_fontsize_dim }
1401         { \l__scripture_chap_fontsize_dim }
1402       \selectfont
1403       \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
1404         { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
1405       \__scripture_select_language:n { chapter }
1406       \__scripture_chap_format:n {#1}
1407     }
1408   \dim_gset:Nn \g__scripture_chap_width_dim
1409     { \box_wd:N \l__scripture_chap_tmp_box }

```

Avoid breaks in the middle of drop chapters.

```

1410   \int_set_eq:NN \l__scripture_clubpenalty_saved_int \clubpenalty
1411   \clubpenalty 10000
1412   \bool_if:NF \l__scripture_verse_para_bool
1413   {
1414     \str_if_eq:VnTF \l__scripture_currenvir_str { poetry }
1415     {
1416       \int_gset_eq:NN
1417         \g__scripture_current_group_level_b_int \currentgrouplevel
1418     }
1419     {
1420       \int_gset:Nn \g__scripture_current_group_level_b_int
1421         { \currentgrouplevel + 1 }
1422     }
1423     \__scripture_chap_par_save_prevgraf_at_group_level:
1424   }
1425 }

```

(End of definition for __scripture_drop_chap_set_up:n.)

`_scripture_drop_chap_output:` Output a drop chapter.

```

1426 \cs_new_protected:Nn \__scripture_drop_chap_output:
1427 {
1428   \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool

```



```

1429 \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
1430 {
1431   \addvspace
1432   {
1433     \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
1434   }
1435 }
1436 \cs_if_eq:NNTF \noindent \__scripture_para_noindent:
1437 {
1438   \legacy_if:nTF { @newlist }
1439   {
1440     \skip_horizontal:n { - \l__scripture_parindent_tl }
1441   }
1442   { \__scripture_noindent_saved: }
1443   \__scripture_pilcrow_output:
1444 }
1445 { \noindent }
1446 \parshape 3 ~
1447 \dim_eval:n
1448 {
1449   \@totalleftmargin +
1450   \g__scripture_chap_width_dim +
1451   \l__scripture_chap_sep_tl
1452 } ~
1453 \dim_eval:n
1454 {
1455   \linewidth -
1456   \g__scripture_chap_width_dim -
1457   \l__scripture_chap_sep_tl
1458 } ~
1459 \dim_eval:n
1460 {
1461   \@totalleftmargin +
1462   \g__scripture_chap_width_dim +
1463   \l__scripture_chap_sep_tl
1464 } ~
1465 \dim_eval:n
1466 {
1467   \linewidth -
1468   \g__scripture_chap_width_dim -
1469   \l__scripture_chap_sep_tl
1470 } ~
1471 \@totalleftmargin ~
1472 \linewidth
1473 \group_begin:
1474 \hook_use:n { scripture / chap / before }
1475 \box_use:N \l__scripture_chap_box
1476 \hook_use:n { scripture / chap / after }
1477 \group_end:
1478 \bool_if:NTF \l__scripture_chap_show_verse_bool
1479 {
1480   \__scripture_verse_output:n { 1 }
1481 }
1482 { \__scripture_insert_verse_mark: }

```

```

1483 \bool_if:NT \l__scripture_verse_para_bool
1484 {
1485   \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
1486   \__scripture_chap_par_save_prevgraf_at_group_level:
1487 }
1488 \hook_gc_clear_next_code:n { scripture / pilcrow }
1489 }

```

(End of definition for __scripture_drop_chap_output:.)

__scripture_nodrop_chap:n Format a non-drop chapter.

```

1490 \cs_new_protected:Nn \__scripture_nodrop_chap:n
1491 {
1492   \tl_gset:Nn \scripturecurrentchapter {#1}
1493   \tl_gset:Nn \scripturecurrentverse { 1 }
1494   \group_begin:
1495   \bool_if:NT \l__scripture_verse_para_bool
1496   {
1497     \legacy_if:nF { @newlist }
1498     {
1499       \mode_if_vertical:F { \para_end: }
1500     }
1501   }
1502   \bool_if:NTF \l__scripture_chap_show_bool
1503   {
1504     \__scripture_get_X_height:
1505     \hcoffin_set:Nn \l__scripture_nodrop_chap_coffin
1506     {
1507       \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
1508       { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
1509       \l__scripture_chap_font_tl
1510       \__scripture_chap_format:n {#1}
1511       \kern \l__scripture_chap_sep_tl
1512     }
1513     \dim_zero:N \l__scripture_nodrop_chap_voffset_dim
1514     \tl_if_eq:NnT \l__scripture_chapter_valign_tl { t }
1515     {
1516       \dim_set_eq:NN
1517       \l__scripture_nodrop_chap_voffset_dim
1518       \l__scripture_chap_X_height_dim
1519     }
1520     \tl_if_eq:NnT \l__scripture_chapter_valign_tl { vc }
1521     {
1522       \dim_set:Nn \l__scripture_nodrop_chap_voffset_dim
1523       { 0.5 \l__scripture_chap_X_height_dim }
1524     }
1525     \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
1526     {
1527       \addvspace
1528       {
1529         \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
1530       }
1531     }
1532     \hook_use:n { scripture / chap / before }

```

```

1533 \bool_if:NTF \l__scripture_chap_smash_bool
1534 {
1535   \mode_leave_vertical:
1536   \smash
1537   {
1538     \exp_args:NNoo
1539     \coffin_typeset:Nnnnn \l__scripture_nodrop_chap_coffin
1540     { \l__scripture_chapter_align_tl }
1541     { \l__scripture_chapter_valign_tl }
1542     { Opt }
1543     { \l__scripture_nodrop_chap_voffset_dim }
1544   }
1545 }
1546 {
1547   \exp_args:NNoo
1548   \coffin_typeset:Nnnnn \l__scripture_nodrop_chap_coffin
1549   { \l__scripture_chapter_align_tl }
1550   { \l__scripture_chapter_valign_tl }
1551   { Opt }
1552   { \l__scripture_nodrop_chap_voffset_dim }
1553 }
1554 \nobreak
1555 \skip_horizontal:N \c_zero_skip
1556 \hook_use:n { scripture / chap / after }
1557 \bool_if:NTF \l__scripture_chap_show_verse_bool
1558 {
1559   \__scripture_verse_output:n { 1 }
1560 }
1561 { \__scripture_insert_verse_mark: }
1562 \str_if_eq:VnF \l__scripture_currenvir_str { midparachap }
1563 { \__scripture_pilcrow_output: }
1564 }
1565 {
1566   \hook_use:n { scripture / chap / before }
1567   \hook_use:n { scripture / chap / after }
1568   \vs { 1 }
1569 }
1570 \group_end:
1571 \parshape 1 ~ \@totalleftmargin ~ \linewidth
1572 \hook_gc_clear_next_code:n { scripture / pilcrow }
1573 }

```

(End of definition for __scripture_nodrop_chap:n.)

__scripture_para_chap:n Format a chapter in its own paragraph.

```

1574 \cs_new_protected:Nn \__scripture_para_chap:n
1575 {
1576   \tl_gset:Nn \scripturecurrentchapter {#1}
1577   \tl_gset:Nn \scripturecurrentverse { 1 }
1578   \group_begin:
1579   \skip_zero:N \parskip
1580   \bool_if:NTF \l__scripture_chap_show_bool
1581   {
1582     \dim_set_eq:NN \l__scripture_para_chap_indent_dim \linewidth

```

```

1583 \str_if_eq:VnTF \l__scripture_currenvir_str { scripture }
1584 {
1585     \dim_set:Nn \l__scripture_para_chap_indent_dim
1586     { 0.5 \l__scripture_para_chap_indent_dim }
1587 }
1588 {
1589     \dim_add:Nn \l__scripture_para_chap_indent_dim
1590     { \leftmargin + \rightmargin }
1591     \dim_set:Nn \l__scripture_para_chap_indent_dim
1592     { 0.5 \l__scripture_para_chap_indent_dim }
1593     \dim_sub:Nn \l__scripture_para_chap_indent_dim { \leftmargin }
1594 }
1595 \mode_if_vertical:F { \para_end: }
1596 \addvspace { \l__scripture_chap_para_aboveskip_tl }
1597 \bool_gset_eq:NN \g_tmpa_bool \g__scripture_suppress_next_pilcrow_bool
1598 \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1599 \str_if_eq:VnTF \l__scripture_currenvir_str { midparachap }
1600 { \__scripture_noindent_saved: }
1601 { \noindent }
1602 \bool_gset_eq:NN \g__scripture_suppress_next_pilcrow_bool \g_tmpa_bool
1603 \skip_horizontal:N \l__scripture_para_chap_indent_dim
1604 \hbox_overlap_center:n
1605 {
1606     \hook_use:n { scripture / chap / before }
1607     \group_begin:
1608     \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
1609     { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
1610     \l__scripture_chap_font_tl
1611     \__scripture_select_language:n { chapter }
1612     \__scripture_chap_format:n {#1}
1613     \group_end:
1614     \hook_use:n { scripture / chap / after }
1615 }
1616 \para_end:
1617 \nobreak
1618 \bool_lazy_any:nTF
1619 {
1620     { \str_if_eq_p:Vn \l__scripture_currenvir_str { scripture } }
1621     { \str_if_eq_p:Vn \l__scripture_currenvir_str { midparachap } }
1622     { \str_if_eq_p:Vn \l__scripture_currenvir_str { narrow } }
1623 }
1624 {
1625     \legacy_if_set:nn
1626     { @afterindent } { \l__scripture_chap_para_afterindent_bool }
1627 }
1628 {
1629     \legacy_if_set_true:n { @afterindent }
1630 }
1631 \@afterheading
1632 }
1633 {
1634     \hook_use:n { scripture / chap / before }
1635     \hook_use:n { scripture / chap / after }
1636 }

```

```

1637 \peek_meaning:NTF \c_space_token
1638 {
1639   \addvspace { \l__scripture_chap_para_belowskip_tl }
1640   \group_end:
1641   \hook_use:n { scripture / poetry / pilcrow }
1642 }
1643 {
1644   \addvspace { \l__scripture_chap_para_belowskip_tl }
1645   \cs_if_eq:NNTF \vs \__scripture_poetry_mode_vertical_verse:n
1646   {
1647     \bool_gset_eq:NN
1648     \g_tmpa_bool \g__scripture_suppress_next_pilcrow_bool
1649     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1650     \noindent
1651     \vs { 1 }
1652     \bool_gset_eq:NN
1653     \g__scripture_suppress_next_pilcrow_bool \g_tmpa_bool
1654   }
1655   {
1656     \str_if_eq:VnT \l__scripture_currenvir_str { midparachap }
1657     { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
1658     \vs { 1 }
1659   }
1660   \group_end:
1661   \hook_use:n { scripture / poetry / pilcrow }
1662 }
1663 }

```

(End of definition for __scripture_para_chap:n.)

__scripture_chap:n Position a drop chapter.

```

1664 \cs_new_protected:Nn \__scripture_chap:n
1665 {
1666   \__scripture_drop_chap_set_up:n {#1}
1667   \hbox_set:Nn \l__scripture_chap_box
1668   {
1669     \box_move_down:nn
1670     { \baselineskip }
1671     {
1672       \hbox_overlap_left:n
1673       {
1674         \box_use:N \l__scripture_chap_tmp_box
1675         \skip_horizontal:N \l__scripture_chap_sep_tl
1676       }
1677     }
1678   }
1679   \box_set_ht:Nn \l__scripture_chap_box { \c_zero_dim }
1680   \box_set_dp:Nn \l__scripture_chap_box { \c_zero_dim }
1681   \bool_if:NTF \l__scripture_mid_para_chap_bool
1682   { \__scripture_mid_para_drop_chap_output: }
1683   {
1684     \bool_if:NTF \l__scripture_verse_para_bool
1685     { \para_end: }
1686     { \par }

```

```

1687     \__scripture_drop_chap_output:
1688     \str_if_eq:VnF \l__scripture_currenvir_str { midparachap }
1689     { \__scripture_pilcrow_output: }
1690   }
1691   \int_gset_eq:NN \g__scripture_current_group_level_a_int \currentgrouplevel
1692   \__scripture_reset_par_at_group_level:
1693 }

```

(End of definition for __scripture_chap:n.)

`\g__scripture_current_group_level_a_int` Track the group level so indent and spacing is correctly reset after a drop chapter even
`\g__scripture_current_group_level_c_int` if the first paragraph of the chapter contains something like a `\footnote`.

```

1694 \int_new:N \g__scripture_current_group_level_a_int
1695 \int_new:N \g__scripture_current_group_level_b_int
1696 \int_new:N \g__scripture_current_group_level_c_int

```

(End of definition for \g__scripture_current_group_level_a_int and \g__scripture_current_group_level_c_int.)

`__scripture_reset_par_at_group_level:` This function skips over inner groups (like `\footnote`) before resetting the `\parshape` after a drop chapter.

```

1697 \cs_new_protected:Nn \__scripture_reset_par_at_group_level:
1698 {
1699   \hook_gput_next_code:nn { para / before }
1700   {
1701     \int_compare:nNnTF
1702     { \g__scripture_current_group_level_a_int } = { \currentgrouplevel }
1703     {
1704       \legacy_if:nF { @newlist }
1705       {
1706         \int_compare:nNnT
1707         { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
1708         {
1709           \dim_compare:nNnT { \parskip } < { \baselineskip }
1710           {
1711             \dim_compare:nNnT
1712             {
1713               \g__scripture_chap_width_dim +
1714               \l__scripture_chap_sep_tl
1715             }
1716             >
1717             { \parindent }
1718             {
1719               \hook_gput_next_code:nn { para / begin }
1720               {
1721                 \para_omit_indent:
1722                 \skip_horizontal:n
1723                 {
1724                   \g__scripture_chap_width_dim +
1725                   \l__scripture_chap_sep_tl
1726                 }
1727               }
1728             }
1729           }
1730         }
1731       }
1732     }
1733   }

```

```

1730         }
1731         \int_gzero:N \g__scripture_chap_par_prevgraf_int
1732     }
1733     \parshape 1 ~ \@totalleftmargin ~ \linewidth
1734 }
1735 { \__scripture_reset_par_at_group_level: }
1736 }
1737 }

```

(End of definition for __scripture_reset_par_at_group_level:.)

ture_chap_par_save_prevgraf_at_group_level: This function skips over inner groups (like \footnote) before saving the \prevgraf.

```

1738 \cs_new_protected:Nn \__scripture_chap_par_save_prevgraf_at_group_level:
1739 {
1740     \hook_gput_next_code:nn { para / after }
1741     {
1742         \int_compare:nNnTF
1743         { \g__scripture_current_group_level_b_int } = { \currentgrouplevel }
1744         {
1745             \int_gset_eq:NN \g__scripture_chap_par_prevgraf_int \prevgraf
1746             \int_compare:nNnT
1747             { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
1748             { \penalty 10000 }
1749             \hook_gput_next_code:nn { para / before }
1750             {
1751                 \int_set_eq:NN
1752                 \clubpenalty \l__scripture_clubpenalty_saved_int
1753             }
1754         }
1755         { \__scripture_chap_par_save_prevgraf_at_group_level: }
1756     }
1757 }

```

(End of definition for __scripture_chap_par_save_prevgraf_at_group_level:.)

ture_zero_chap_par_prevgraf_at_group_level: This function skips over inner groups (like \footnote) before zeroing \g__scripture_chap_par_prevgraf_int.

```

1758 \cs_new_protected:Nn \__scripture_zero_chap_par_prevgraf_at_group_level:
1759 {
1760     \hook_gput_next_code:nn { para / after }
1761     {
1762         \int_compare:nNnTF
1763         { \g__scripture_current_group_level_c_int } = { \currentgrouplevel }
1764         {
1765             \hook_gput_next_code:nn { para / after }
1766             { \int_gzero:N \g__scripture_chap_par_prevgraf_int }
1767         }
1768         { \__scripture_zero_chap_par_prevgraf_at_group_level: }
1769     }
1770 }

```

(End of definition for __scripture_zero_chap_par_prevgraf_at_group_level:.)

3.5.5 Formatting and printing verse numbers

<code>\l_scripture_verse_par_start_bool</code>	Flag for whether the current verse is at the start of a paragraph. 1771 <code>\bool_new:N \l__scripture_verse_par_start_bool</code> (End of definition for <code>\l_scripture_verse_par_start_bool</code> .)
<code>\l_scripture_no_verse_first_bool</code>	Suppress the next verse from formatting differently if it begins a new paragraph. 1772 <code>\bool_new:N \l__scripture_no_verse_first_bool</code> (End of definition for <code>\l_scripture_no_verse_first_bool</code> .)
<code>\g_scripture_suppress_next_verse_para_bool</code>	Suppress the next verse from starting a new paragraph when verse/para is in use. 1773 <code>\bool_new:N \g__scripture_suppress_next_verse_para_bool</code> (End of definition for <code>\g_scripture_suppress_next_verse_para_bool</code> .)
<code>__scripture_verse_output:n</code>	Output a formatted verse. 1774 <code>\cs_new_protected:Nn __scripture_verse_output:n</code> 1775 <code>{</code> 1776 <code>\tl_gset:Nn \scripturecurrentverse {#1}</code> 1777 <code>\legacy_if:nTF { @endpe } {</code> 1778 <code>\bool_set_false:N \l__scripture_verse_par_start_bool } {</code> 1779 <code>\bool_set_true:N \l__scripture_verse_par_start_bool }</code> 1780 <code>\mode_if_vertical:TF { \mode_leave_vertical: } {</code> 1781 <code>{</code> 1782 <code>\bool_set_false:N \l__scripture_verse_par_start_bool</code> 1783 <code>\skip_if_eq:nnF { \l__scripture_verse_presep_tl } { \c_zero_skip }</code> 1784 <code>{</code> 1785 <code>\tl_if_eq:nnF {#1} { 1 } {</code> 1786 <code>{</code> 1787 <code>\skip_horizontal:n { \l__scripture_verse_presep_tl }</code> 1788 <code>}</code> 1789 <code>}</code> 1790 <code>}</code> 1791 <code>}</code> 1792 <code>\bool_if:NT \l__scripture_no_verse_first_bool {</code> 1793 <code>\bool_set_false:N \l__scripture_verse_par_start_bool</code> 1794 <code>\bool_set_false:N \l__scripture_no_verse_first_bool</code> 1795 <code>}</code> 1796 <code>\group_begin:</code> 1797 <code>\bool_if:NT \l__scripture_verse_para_bool {</code> 1798 <code>{</code> 1799 <code>\box_if_empty:NF \g__scripture_post_env_indent_box</code> 1800 <code>{ \box_use_drop:N \g__scripture_post_env_indent_box }</code> 1801 <code>}</code> 1802 <code>\hook_use:n { scripture / verse / before }</code> 1803 <code>\bool_if:NT \l__scripture_verse_show_bool {</code> 1804 <code>{</code> 1805 <code>\group_begin:</code> 1806 <code>\l__scripture_verse_font_tl</code> 1807 <code>\str_if_eq:VnF \l__scripture_verse_colour_tl { . }</code> 1808 <code>{ \exp_args:NV \color_select:n \l__scripture_verse_colour_tl }</code> 1809 <code>__scripture_select_language:n { verse }</code> 1810 <code>}</code>


```

1811 \bool_lazy_all:nTF
1812 {
1813   { \l__scripture_verse_first_bool }
1814   { \l__scripture_verse_par_start_bool }
1815   {
1816     \bool_not_p:n
1817     {
1818       \str_if_eq_p:Vn \l__scripture_currenvir_str { poetry }
1819     }
1820   }
1821 }
1822 {
1823   \__scripture_verse_first_format:n {#1}
1824   \group_end:
1825   \kern \l__scripture_verse_first_sep_tl
1826 }
1827 {
1828   \__scripture_verse_format:n {#1}
1829   \group_end:
1830   \kern \l__scripture_verse_sep_tl
1831 }
1832 \nobreak
1833 \skip_horizontal:N \c_zero_skip
1834 }
1835 \__scripture_insert_verse_mark:
1836 \hook_use:n { scripture / verse / after }
1837 \hook_use:n { scripture / pilcrow }
1838 \group_end:
1839 }

```

(End of definition for __scripture_verse_output:n.)

_scripture_verse_compact_output:n Output a formatted verse.

```

1840 \cs_new_protected:Nn \__scripture_verse_compact_output:n
1841 {
1842   \mode_if_vertical:T
1843   { \__scripture_set_pilcrow_hook: }
1844   \__scripture_verse_output:n {#1}
1845 }

```

(End of definition for __scripture_verse_compact_output:n.)

_scripture_verse_para_output:n Output a formatted verse when **verse/para** is set.

```

1846 \cs_new_protected:Nn \__scripture_verse_para_output:n
1847 {
1848   \mode_if_vertical:T
1849   { \__scripture_set_pilcrow_hook: }
1850   \hook_if_empty:nT { scripture / pilcrow }
1851   {
1852     \dim_set:Nn \parindent { \l__scripture_para_indent_tl }
1853   }
1854   \bool_if:NTF \g__scripture_suppress_next_verse_para_bool
1855   { \bool_gset_false:N \g__scripture_suppress_next_verse_para_bool }
1856   {
1857     \legacy_if:nF { @newlist }

```

```

1858         {
1859             \mode_if_vertical:F
1860             {
1861                 \para_end:
1862                 \parshape 1 ~ \@totalleftmargin ~ \linewidth
1863             }
1864         }
1865     }
1866     \__scripture_verse_output:n {#1}
1867 }

```

(End of definition for __scripture_verse_para_output:n.)

__scripture_hanging_verse_para:n Output a formatted verse when **verse/para** is set and in a hanging environment.

```

1868 \cs_new_protected:Nn \__scripture_hanging_verse_para:n
1869 {
1870     \mode_if_vertical:T
1871     { \__scripture_set_pilcrow_hook: }
1872     \bool_if:NF \g__scripture_suppress_next_verse_para_bool
1873     {
1874         \legacy_if:nF { @newlist }
1875         {
1876             \mode_if_vertical:F
1877             {
1878                 \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1879                 \hook_gput_next_code:nn { para / begin }
1880                 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
1881                 \par
1882                 \__scripture_noindent_saved:
1883             }
1884         }
1885     }
1886     \bool_gset_false:N \g__scripture_suppress_next_verse_para_bool
1887     \__scripture_hanging_verse:n {#1}
1888 }

```

(End of definition for __scripture_hanging_verse_para:n.)

3.5.6 Printing pilcrows

\g__scripture_suppress_next_pilcrow_bool Flag to suppress the next pilcrow.

```

1889 \bool_new:N \g__scripture_suppress_next_pilcrow_bool

```

(End of definition for \g__scripture_suppress_next_pilcrow_bool.)

__scripture_set_pilcrow_hook: Set the next **scripture/pilcrow** hook to output a pilcrow providing the hook is currently empty. **verse/para** or **compact** options are set.

```

1890 \cs_new_protected:Nn \__scripture_set_pilcrow_hook:
1891 {
1892     \hook_if_empty:nT { scripture / pilcrow }
1893     {
1894         \hook_gput_next_code:nn
1895         { scripture / pilcrow } { \__scripture_pilcrow_output: }
1896     }
1897 }

```

(End of definition for `__scripture_set_pilcrow_hook:`)

`__scripture_pilcrow_output:` Output a pilcrow (¶). Pilrows only display when either the **verse/para** or **compact** options are set.

```

1898 \cs_new_protected:Nn \__scripture_pilcrow_output:
1899 {
1900   \bool_gset_false:N \g__scripture_para_mode_vertical_bool
1901   \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
1902   \bool_if:NT \l__scripture_pilcrow_bool
1903   {
1904     \bool_lazy_or:nnT
1905     { \l__scripture_verse_para_bool } { \l__scripture_compact_bool }
1906     {
1907       \bool_if:NTF \g__scripture_suppress_next_pilcrow_bool
1908       { \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool }
1909       {
1910         \P
1911         \kern \l__scripture_pilcrow_sep_tl
1912         \nobreak
1913         \skip_horizontal:N \c_zero_skip
1914       }
1915     }
1916   }
1917 }
```

(End of definition for `__scripture_pilcrow_output:`)

3.5.7 Calculating the length of the final line in a paragraph

`\g__scripture_final_line_dim` Store the length of the final line in a paragraph.

```

1918 \dim_new:N \g__scripture_final_line_dim
```

(End of definition for `\g__scripture_final_line_dim:`)

`__scripture_calc_final_line_length:` Calculate the length of the final line in a paragraph.

```

1919 \cs_new_protected:Nn \__scripture_calc_final_line_length:
1920 {
1921   \skip_set:Nn \abovedisplayshortskip { - \baselineskip }
1922   \skip_set_eq:NN \abovedisplayskip \abovedisplayshortskip
1923   \skip_zero:N \belowdisplayshortskip
1924   \skip_zero:N \belowdisplayskip
1925   $$
1926   \hook_use:n { para / after }
1927   \dim_compare:nNnT { \predisplaysize } > { \c_zero_dim }
1928   {
1929     \dim_gset:Nn \g__scripture_final_line_dim
1930     { \predisplaysize - 2em }
1931   }
1932   \cs_gset_nopar:Nx \__scripture_reset_spacing:
1933   {
1934     \dim_set:Nn \predisplaysize { \dim_use:N \predisplaysize }
1935     \int_set:Nn \prevgraf { \int_use:N \prevgraf }
1936     \int_gset:Nn \g__scripture_chap_par_prevgraf_int
1937     { \int_use:N \g__scripture_chap_par_prevgraf_int }
```

```

1938     }
1939     $$
1940     \__scripture_reset_spacing:
1941     \dim_compare:nNnT { \g__scripture_final_line_dim } > { \paperwidth }
1942     {
1943         \dim_gset:Nn \g__scripture_final_line_dim
1944         { \totalleftmargin + \linewidth }
1945     }
1946     \dim_compare:nNnT { \g__scripture_final_line_dim } > { \c_zero_dim - 1sp }
1947     {
1948         \skip_vertical:n { - \baselineskip - \parskip }
1949     }
1950 }

```

(End of definition for __scripture_calc_final_line_length:.)

3.5.8 The **scripture** environment

`\l__scripture_currenvir_str` Store the current **scripture** environment.

```

1951 \str_new:N \l__scripture_currenvir_str

```

(End of definition for \l__scripture_currenvir_str.)

`\l__scripture_ref_tl` Store the reference passed to the **scripture** environment.

```

1952 \tl_new:N \l__scripture_ref_tl

```

(End of definition for \l__scripture_ref_tl.)

`__scripture_added:n` Mark text as added.

```

1953 \cs_new_protected_nopar:Nn \__scripture_added:n
1954 {
1955     \group_begin:
1956     \l__scripture_added_font_tl
1957     \__scripture_added_format:n {#1}
1958     \group_end:
1959 }

```

(End of definition for __scripture_added:n.)

`__scripture_heading:n` Output a non-canonical heading.

```

1960 \cs_new_protected_nopar:Nn \__scripture_heading:n
1961 {
1962     \str_if_eq:VnTF \l__scripture_currenvir_str { scripture }
1963     {
1964         \bool_if:NTF \l__scripture_heading_show_bool
1965         {
1966             \parshape 1 ~ \totalleftmargin ~ \linewidth
1967             \group_begin:
1968             \bool_set_false:N \l__scripture_pilcrow_bool
1969             \legacy_if:nTF { @newlist }
1970             {
1971                 \skip_horizontal:n { - \parindent }
1972             }
1973             {
1974                 \__scripture_list_para_end:

```

```

1975         \addvspace { \l__scripture_heading_aboveskip_tl }
1976         \__scripture_noindent_saved:
1977     }
1978     \l__scripture_heading_align_tl
1979     \l__scripture_heading_font_tl
1980     \__scripture_heading_format:n {#1}
1981     \mark_insert:nn { scripture / heading } {#1}
1982     \__scripture_list_para_end:
1983     \group_end:
1984     \bool_if:NTF \l__scripture_verseline_para_bool
1985     {
1986         \cs_set_eq:NN \noindent \__scripture_para_noindent:
1987     }
1988     {
1989         \cs_set_eq:NN \noindent \__scripture_noindent_saved:
1990     }
1991     \legacy_if_set:nn
1992     { @afterindent } { \l__scripture_heading_afterindent_bool }
1993     \bool_if:NF \l__scripture_heading_afterindent_bool
1994     {
1995         \hook_gput_next_code:nn { para / before }
1996         { \para_omit_indent: }
1997     }
1998     \nobreak
1999     \@afterheading
2000     \addvspace { \l__scripture_heading_belowskip_tl }
2001 }
2002 {
2003     \mark_insert:nn { scripture / heading } {#1}
2004     \ignorespaces
2005 }
2006 }
2007 {
2008     \msg_error:nnx
2009     { scripture }
2010     { heading-error }
2011     { \l__scripture_currenvir_str }
2012 }
2013 }

```

(End of definition for __scripture_heading:n.)

__scripture_name:n Output divine name.

```

2014 \cs_new_protected_nopar:Nn \__scripture_name:n
2015 {
2016     \group_begin:
2017     \l__scripture_name_font_tl
2018     \__scripture_name_format:n {#1}
2019     \group_end:
2020 }

```

(End of definition for __scripture_name:n.)

__scripture_no_first_verseline: Suppress special formatting for first verse if active.

```

2021 \cs_new_protected_nopar:Nn \__scripture_no_first_verse:
2022 {
2023   \bool_set_true:N \l__scripture_no_verse_first_bool
2024 }

```

(End of definition for __scripture_no_first_verse:.)

__scripture_ch_output:nn Print a chapter number in the main scripture environment.

```

2025 \cs_new_protected:Nn \__scripture_ch_output:nn
2026 {
2027   \bool_if:NTF \l__scripture_chap_show_bool
2028   {
2029     \bool_if:NTF \l__scripture_chap_drop_bool
2030     {
2031       \__scripture_chap:n {#2}
2032       \bool_lazy_and:nnTF
2033       {#1}
2034       { \bool_not_p:n { \l__scripture_compact_bool } }
2035       {
2036         \hook_gput_next_code:nn { para / after }
2037         {
2038           \int_gzero:N \g__scripture_chap_par_prevgraf_int
2039           \int_gzero:N \g__scripture_mid_para_chap_prevgraf_int
2040         }
2041       }
2042     }
2043     \hook_gput_next_code:nn { para / after }
2044     {
2045       \int_gset_eq:NN
2046       \g__scripture_mid_para_chap_prevgraf_int \prevgraf
2047     }
2048   }
2049   {
2050     \bool_if:NTF \l__scripture_chap_para_bool
2051     { \__scripture_para_chap:n {#2} }
2052     { \__scripture_nodrop_chap:n {#2} }
2053   }
2054 }
2055 {
2056   \bool_if:NTF \l__scripture_chap_para_bool
2057   { \__scripture_para_chap:n {#2} }
2058   { \__scripture_nodrop_chap:n {#2} }
2059 }
2060 }
2061 }

```

(End of definition for __scripture_ch_output:nn.)

__scripture_begin:nn Begin a scripture quotation.

```

2062 \cs_new_protected:Nn \__scripture_begin:nn
2063 {
2064   \cs_set_eq:NN \__scripture_noindent_saved: \noindent
2065   \bool_if:NT \l__scripture_active_bool
2066   {
2067     \msg_error:nn { scripture } { nested-environment }

```

```

2068     }
2069     \bool_set_true:N \l__scripture_active_bool
2070     \tl_if_novalue:nF {#1}
2071     { \tl_set:Nn \l__scripture_ref_tl {#1} }
2072     \str_set:Nn \l__scripture_currenvir_str { scripture }
2073     \bool_gset_false:N \g__scripture_first_verse_set_bool
2074     \int_gset:Nn \g__scripture_chapter_int { 1 }
2075     \int_gset:Nn \g__scripture_verse_int { 1 }
2076     \tl_gclear:N \g__scripture_book_tl
2077     \tl_if_novalue:nF {#2}
2078     {
2079         \keys_set:nn { scripture } {#2}
2080     }
2081     \tl_gset:NV \scripturecurrentbook \g__scripture_book_tl
2082     \tl_gset:Ne \scripturecurrentchapter
2083         { \int_use:N \g__scripture_chapter_int }
2084     \tl_gset:Ne \scripturecurrentverse
2085         { \int_use:N \g__scripture_verse_int }
2086     \l__scripture_before_tl
2087     \__scripture_select_language:n { main }
2088     \int_gzero:N \g__scripture_chap_par_prevgraf_int
Empty \everydisplay to ensure it isn't interrering with spacing.
2089     \everydisplay { }
2090     \int_set:Nn \predisplayskip { 10000 }
2091     \int_set_eq:NN \postdisplayskip \@lowpenalty
2092     \int_set:Nn \@beginparpenalty { - \@lowpenalty }
2093     \int_set_eq:NN \@endparpenalty \@lowpenalty
2094     \bool_gset_false:N \g__scripture_poetry_end_bool
2095     \cs_set_eq:NN \extraskip \__scripture_extra_skip:
2096     \cs_set_eq:NN \redletteron \__scripture_red_letter_on:
2097     \cs_set_eq:NN \redletteroff \__scripture_red_letter_off:
2098     \cs_set_protected:Npn \nopilcrow
2099         { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2100     \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
2101     \bool_gset_false:N \g__scripture_suppress_next_verse_para_bool
2102     \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
2103     \box_gclear:N \g__scripture_post_env_indent_box
2104     \skip_gzero:N \g__scripture_prev_inner_below_skip
2105     \bool_if:NT \l__scripture_chap_para_bool
2106         { \bool_set_false:N \l__scripture_chap_drop_bool }
2107     \DeclareDocumentCommand { \ch } { s o m }
2108     {
2109         \tl_if_novalue:nF {##2}
2110         {
2111             \tl_set:Nn \l__scripture_X_char_tl {##2}
2112         }
2113         \__scripture_ch_output:nn {##1} {##3}
2114     }
2115     \cs_set_eq:NN \vs \__scripture_verse_output:n
2116     \DeclareDocumentCommand { \newch } { s o }
2117     {
2118         \bool_if:NTF \g__scripture_first_verse_set_bool
2119         {
2120             \int_gincr:N \g__scripture_chapter_int

```

```

2121         \int_gset:Nn \g__scripture_verse_int { 1 }
2122     }
2123     { \bool_gset_true:N \g__scripture_first_verse_set_bool }
2124     \bool_if:nTF {##1}
2125     {
2126         \ch* [ ##2 ] { \int_use:N \g__scripture_chapter_int }
2127     }
2128     {
2129         \ch [ ##2 ] { \int_use:N \g__scripture_chapter_int }
2130     }
2131 }
2132 \DeclareDocumentCommand { \newvs } { }
2133 {
2134     \bool_if:NTF \g__scripture_first_verse_set_bool
2135     { \int_gincr:N \g__scripture_verse_int }
2136     { \bool_gset_true:N \g__scripture_first_verse_set_bool }
2137     \vs { \int_use:N \g__scripture_verse_int }
2138 }
2139 \cs_set_eq:NN \added \__scripture_added:n
2140 \cs_set_eq:NN \heading \__scripture_heading:n
2141 \cs_set_eq:NN \name \__scripture_name:n
2142 \cs_set_protected_nopar:Npn \LORD { \name { Lord } }
2143 \cs_set_protected_nopar:Npn \GOD { \name { God } }
2144 \cs_set_eq:NN \nofirstverse \__scripture_no_first_verse:
2145 \cs_set_protected_nopar:Npn \textright ##1
2146 { \__scripture_text_right:nn { \l__scripture_text_right_sep_tl } {##1} }
2147 \hook_gclear_next_code:n { para / before }
2148 \hook_gclear_next_code:n { scripture / pilcow }
2149 \hook_gclear_next_code:n { scripture / poetry / pilcow }
2150 \hook_gput_code:nnn { cmd / @makefntext / before } { scripture }
2151 { \cs_set_eq:NN \noindent \__scripture_noindent_saved: }
2152 \bool_if:NTF \l__scripture_inline_bool
2153 { \l__scripture_inline_font_tl }
2154 { \l__scripture_font_tl }
2155 \skip_if_eq:nnF { \l__scripture_verse_presep_tl } { \c_zero_skip }
2156 {
2157     \bool_set_true:N \l__scripture_verse_first_bool
2158     \cs_set_eq:NN
2159         \__scripture_verse_first_format:n \__scripture_verse_format:n
2160 }
2161 \tl_set:Ne \l__scripture_parindent_tl
2162 { \dim_eval:n { \l__scripture_parindent_tl } }
2163 \exp_args:NnV
2164 \color_set:nn { scripture default colour } \l__scripture_colour_tl
2165 \str_if_eq:VnF \l__scripture_colour_tl { . }
2166 { \color_select:n { scripture default colour } }
2167 \bool_if:NTF \l__scripture_inline_bool
2168 { \l__scripture_inline_begin_tl }
2169 {
2170     \legacy_if:nTF { @newlist }
2171     { \dim_set_eq:NN \l__scripture_outer_itemindent_dim \itemindent }
2172     { \dim_zero:N \l__scripture_outer_itemindent_dim }
2173     \list { }
2174     {

```



```

2175 \dim_set:Nn \leftmargin { \l__scripture_leftmargin_tl }
2176 \dim_set:Nn \rightmargin { \l__scripture_rightmargin_tl }
2177 \dim_set:Nn \itemindent
2178 {
2179   \l__scripture_parindent_tl - \l__scripture_outer_itemindent_dim
2180 }
2181 \dim_set:Nn \listparindent { \l__scripture_parindent_tl }
2182 \skip_zero:N \partopsep
2183 \skip_set:Nn \parsep { \l__scripture_parskip_tl }
2184 \dim_compare:nNnTF { \parskip } > { \l__scripture_aboveskip_tl }
2185 { \skip_zero:N \topsep }
2186 {
2187   \skip_set:Nn \topsep
2188   {
2189     \__scripture_skip_diff:NN
2190     \l__scripture_aboveskip_tl \parskip
2191   }
2192 }
2193 \__scripture_setup_list_noindent:n { \l__scripture_parindent_tl }
2194 \bool_if:NF \l__scripture_indent_bool
2195 {
2196   \hook_gput_next_code:nn { para / begin }
2197   {
2198     \bool_if:NTF \l__scripture_compact_bool
2199     { \para_omit_indent: }
2200     {
2201       \bool_lazy_any:nT
2202       {
2203         {
2204           \str_if_eq_p:Vn
2205           \l__scripture_currenvir_str { midparachap }
2206         }
2207         {
2208           \str_if_eq_p:Vn
2209           \l__scripture_currenvir_str { narrow }
2210         }
2211         {
2212           \str_if_eq_p:Vn
2213           \l__scripture_currenvir_str { scripture }
2214         }
2215       }
2216       {
2217         \legacy_if:nTF { @noparlist }
2218         {
2219           \bool_lazy_and:nnF
2220           {
2221             \str_if_eq_p:Vn
2222             \l__scripture_currenvir_str { narrow }
2223           }
2224           {
2225             \str_if_eq_p:Vn
2226             \l__scripture_ref_position_str { start }
2227           }
2228         { \noindent }

```

```

2229         }
2230         { \para_omit_indent: }
2231     }
2232 }
2233 }
2234 }
2235 }
2236 \cs_set_eq:NN \__scripture_list_para_end: \@par
2237 \item
2238 \relax
2239 }
2240 }

```

(End of definition for __scripture_begin:nn.)

__scripture_end: End a scripture quotation with a reference.

```

2241 \cs_new_protected:Nn \__scripture_end:
2242 {
2243   \str_if_eq:VnT \l__scripture_ref_position_str { start }
2244   {
2245     \tl_clear:N \l__scripture_ref_tl
2246     \str_if_eq:VnT \l__scripture_version_position_str { withref }
2247     { \tl_clear:N \l__scripture_version_tl }
2248   }
2249   \cs_set_eq:NN \noindent \__scripture_noindent_saved:
2250   \bool_if:NTF \l__scripture_inline_bool
2251   {
2252     \unskip
2253     \__scripture_red_letter_off:
2254     \l__scripture_inline_end_tl
2255     \bool_lazy_or:nnTF
2256     {
2257       \tl_if_novalue_p:n { \l__scripture_ref_tl }
2258     }
2259     { \tl_if_empty_p:N \l__scripture_ref_tl }
2260     {
2261       \tl_if_empty:NF \l__scripture_version_tl
2262       {
2263         \skip_horizontal:N \l__scripture_ref_sep_tl
2264         \__scripture_format_full_ref:nn { } { \l__scripture_version_tl }
2265       }
2266     }
2267     {
2268       \skip_horizontal:N \l__scripture_ref_sep_tl
2269       \__scripture_format_full_ref:nn
2270       { \l__scripture_ref_tl } { \l__scripture_version_tl }
2271     }
2272   }
2273   {
2274     \__scripture_red_letter_off:
2275     \skip_zero:N \parskip
2276     \int_set:Nn \postdisplaypenalty { 10000 }
2277     \__scripture_calc_final_line_length:
2278     \dim_gsub:Nn \g__scripture_final_line_dim { \@totalleftmargin }

```

```

2279 \dim_compare:nNnTF { \@outerparskip } > { \l__scripture_belowskip_tl }
2280 { \skip_zero:N \@topsepadd }
2281 {
2282   \skip_set:Nn \@topsepadd
2283   {
2284     \__scripture_skip_diff:NN
2285     \l__scripture_belowskip_tl \@outerparskip
2286   }
2287 }

```

If a **scripture** quotation ends with an inner environment, remove the below skip of the inner environment.

```

2288 \int_compare:nNnT { \prevgraf } = { \c_zero_int }
2289 {
2290   \skip_vertical:n { - \g__scripture_prev_inner_below_skip }
2291 }
2292 \hook_gclear_next_code:n { para / before }
2293 \hook_gclear_next_code:n { para / after }
2294 \noindent
2295 \bool_lazy_or:nnTF
2296 {
2297   \tl_if_novalue_p:n { \l__scripture_ref_tl }
2298 }
2299 { \tl_if_empty_p:N \l__scripture_ref_tl }
2300 {
2301   \tl_if_empty:NNTF \l__scripture_version_tl
2302   { \hbox:n { } }
2303   { \__scripture_reference:n { } }
2304 }
2305 {
2306   \__scripture_reference:n { \l__scripture_ref_tl }
2307 }
2308 \parshape 1 ~ \@totalleftmargin ~ \linewidth

```

Set `@nparlist` to false to ensure `\topsep` is always added **scripture** environment

```

2309 \legacy_if_set_false:n { @nparlist }
2310 \endlist
2311 }
2312 \hook_gremove_code:nn { cmd / @makefntext / before } { scripture }
2313 \cs_set_eq:NN \noindent \__scripture_noindent_saved:
2314 \l__scripture_after_tl
2315 }

```

(End of definition for `__scripture_end:.`)

3.6 Scripture internal environments

`__scripture_compact_end_inner:` End a compact inner environment

```

2316 \cs_new_protected:Nn \__scripture_compact_end_inner:
2317 {
2318   \hook_gclear_next_code:n { para / before }
2319   \hook_gclear_next_code:n { para / begin }
2320   \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
2321   \str_if_eq:VnT \l__scripture_ref_position_str { start }
2322   {

```

```

2323     \tl_gclear:N \l__scripture_ref_tl
2324   }
2325 \bool_lazy_and:nnT
2326 { \l__scripture_verse_para_bool } { \l__scripture_chap_drop_bool }
2327 {
2328   \cs_set_eq:NN \par \para_end:
2329   \__scripture_calc_final_line_length:
2330   \int_gset_eq:NN
2331     \g__scripture_mid_para_chap_prevgraf_int
2332     \g__scripture_chap_par_prevgraf_int
2333   \__scripture_noindent_saved:
2334   \bool_lazy_and:nnTF
2335     { \bool_if_p:N \l__scripture_compact_bool }
2336     {
2337       \int_compare_p:nNn
2338         { \g__scripture_mid_para_chap_prevgraf_int } = { \c_one_int }
2339     }
2340   {
2341     \hook_gput_next_code:nn
2342       { env / \l__scripture_currenvir_str / after }
2343     {
2344       \parshape 3 ~
2345       \@totalleftmargin ~ \linewidth ~
2346       \dim_eval:n
2347         {
2348           \@totalleftmargin +
2349           \g__scripture_chap_width_dim +
2350           \l__scripture_chap_sep_tl
2351         } ~
2352       \dim_eval:n
2353         {
2354           \linewidth -
2355           \g__scripture_chap_width_dim -
2356           \l__scripture_chap_sep_tl
2357         } ~
2358       \@totalleftmargin ~ \linewidth
2359     }
2360     \hbox_to_wd:nn
2361       {
2362         \g__scripture_final_line_dim -
2363         \g__scripture_chap_width_dim -
2364         \l__scripture_chap_sep_tl
2365       }
2366       { }
2367   }
2368   {
2369     \hbox_to_wd:nn { \g__scripture_final_line_dim } { }
2370   }
2371   \c_space_token
2372   \kern 0pt
2373   \cs_set_eq:NN \par \relax
2374   \int_gset:Nn \g__scripture_current_group_level_a_int
2375     { \currentgrouplevel - \c_one_int }
2376   \__scripture_reset_par_at_group_level:

```

```

2377     }
2378     \unskip
2379 }

```

(End of definition for `__scripture_compact_end_inner:`.)

3.6.1 The `center` environment

`__scripture_center_begin:n` Begin a centred section in a scripture quotation.

```

2380 \cs_new_protected:Nn \__scripture_center_begin:n
2381 {
2382   \bool_if:NT \l__scripture_active_inner_bool
2383   {
2384     \msg_error:nn { scripture } { nested-environment }
2385   }
2386   \bool_set_true:N \l__scripture_active_inner_bool
2387   \str_set:Nn \l__scripture_currenvir_str { center }
2388   \tl_if_novalue:nF {#1}
2389   {
2390     \keys_set:nn
2391       { scripture / center }
2392       {#1}
2393   }
2394   \legacy_if:nTF { @newlist }
2395   {
2396     \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
2397   }
2398   { \dim_zero:N \l__scripture_outer_itemindent_dim }
2399   \cs_set_eq:NN \par \para_end:
2400   \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
2401   \list { }
2402   {
2403     \bool_lazy_or:nnF
2404       { \mode_if_vertical_p: }
2405       { \g__scripture_para_mode_vertical_bool }
2406       { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2407     \dim_zero:N \labelsep
2408     \dim_set:Nn \leftmargin { \l__scripture_center_leftmargin_tl }
2409     \dim_set:Nn \rightmargin { \l__scripture_center_rightmargin_tl }
2410     \dim_set:Nn \itemindent { - \l__scripture_outer_itemindent_dim }
2411     \dim_zero:N \listparindent
2412     \skip_set_eq:NN \parsep \parskip
2413     \skip_zero:N \partopsep
2414     \dim_compare:nNnTF
2415       { \parskip } > { \l__scripture_center_aboveskip_tl }
2416       { \skip_zero:N \topsep }
2417       {
2418         \skip_set:Nn \topsep
2419         {
2420           \__scripture_skip_diff:NN
2421             \l__scripture_center_aboveskip_tl \parskip
2422         }
2423       }
2424     \__scripture_setup_list_noindent:n { \c_zero_dim }

```

```

2425     }
2426     \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2427     { \addvspace { \baselineskip } }
2428     \int_gzero:N \g__scripture_chap_par_prevgraf_int
2429     \centering
2430     \bool_if:NT \l__scripture_verse_para_bool
2431     {
2432         \cs_set_eq:NN \par \__scripture_para_par:
2433         \cs_set_protected:Npn \\\
2434         {
2435             \mode_if_horizontal:TF
2436             { \unskip }
2437             { \@nolnerr }
2438         }
2439     }
2440     \item
2441     \relax
2442 }

```

(End of definition for __scripture_center_begin:n.)

__scripture_center_end: End a centred section in a scripture quotation.

```

2443 \cs_new_protected:Nn \__scripture_center_end:
2444 {
2445     \cs_set_eq:NN \par \para_end:
2446     \dim_gset:Nn \g__scripture_final_line_dim
2447     { \@totalleftmargin + \linewidth }
2448     \dim_compare:nNnTF
2449     { \@outerparskip } > { \l__scripture_center_belowskip_tl }
2450     { \skip_zero:N \@topsepadd }
2451     {
2452         \skip_set:Nn \@topsepadd
2453         {
2454             \__scripture_skip_diff:NN
2455             \l__scripture_center_belowskip_tl \@outerparskip
2456         }
2457     }
2458     \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
2459     \legacy_if_set_false:n { @nparlist }
2460     \endlist
2461     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2462     \hook_gput_next_code:nn { para / begin }
2463     { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2464 }

```

(End of definition for __scripture_center_end:.)

3.6.2 The **flushright** environment

__scripture_flushright_begin:n Begin a flushright section in a scripture quotation.

```

2465 \cs_new_protected:Nn \__scripture_flushright_begin:n
2466 {
2467     \bool_if:NT \l__scripture_active_inner_bool
2468     {

```

```

2469     \msg_error:nn { scripture } { nested-environment }
2470   }
2471   \bool_set_true:N \l__scripture_active_inner_bool
2472   \str_set:Nn \l__scripture_currenvir_str { flushright }
2473   \tl_if_novalue:nF {#1}
2474   {
2475     \keys_set:nn
2476       { scripture / flushright }
2477       {#1}
2478   }
2479   \legacy_if:nTF { @newlist }
2480   {
2481     \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
2482   }
2483   { \dim_zero:N \l__scripture_outer_itemindent_dim }
2484   \cs_set_eq:NN \par \para_end:
2485   \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
2486   \list { }
2487   {
2488     \bool_lazy_or:nnF
2489       { \mode_if_vertical_p: }
2490       { \g__scripture_para_mode_vertical_bool }
2491       { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2492     \dim_zero:N \labelsep
2493     \dim_set:Nn \leftmargin { \l__scripture_flushright_leftmargin_tl }
2494     \dim_set:Nn \rightmargin { \l__scripture_flushright_rightmargin_tl }
2495     \dim_set:Nn \itemindent { - \l__scripture_outer_itemindent_dim }
2496     \dim_zero:N \listparindent
2497     \skip_set_eq:NN \parsep \parskip
2498     \skip_zero:N \partopsep
2499     \dim_compare:nNnTF
2500       { \parskip } > { \l__scripture_flushright_aboveskip_tl }
2501       { \skip_zero:N \topsep }
2502       {
2503         \skip_set:Nn \topsep
2504         {
2505           \__scripture_skip_diff:NN
2506             \l__scripture_flushright_aboveskip_tl \parskip
2507         }
2508       }
2509     \__scripture_setup_list_noindent:n { \c_zero_dim }
2510   }
2511   \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2512   { \addvspace { \baselineskip } }
2513   \int_gzero:N \g__scripture_chap_par_prevgraf_int
2514   \raggedleft
2515   \bool_if:NT \l__scripture_verse_para_bool
2516   {
2517     \cs_set_eq:NN \par \__scripture_para_par:
2518     \cs_set_protected:Npn \\\
2519     {
2520       \mode_if_horizontal:TF
2521       { \unskip }
2522       { \@nolnerr }

```

```

2523     }
2524   }
2525   \item
2526   \relax
2527 }

```

(End of definition for `_scripture_flushright_begin:n`.)

`_scripture_flushright_end:` End a flushright section in a scripture quotation.

```

2528 \cs_new_protected:Nn \_scripture_flushright_end:
2529 {
2530   \cs_set_eq:NN \para \para_end:
2531   \dim_gset:Nn \g__scripture_final_line_dim { \@totalleftmargin + \linewidth }
2532   \dim_compare:nNnTF
2533     { \@outerparskip } > { \l__scripture_flushright_belowskip_tl }
2534     { \skip_zero:N \@topsepadd }
2535     {
2536       \skip_set:Nn \@topsepadd
2537       {
2538         \_scripture_skip_diff:NN
2539         \l__scripture_flushright_belowskip_tl \@outerparskip
2540       }
2541     }
2542   \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
2543   \legacy_if_set_false:n { @nparlist }
2544   \endlist
2545   \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2546   \hook_gput_next_code:nn { para / begin }
2547     { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2548 }

```

(End of definition for `_scripture_flushright_end:.`)

3.6.3 The **hanging** environment

`\l__scripture_hanging_parindent_saved_dim` Save the `\parindent` for later use in **hanging** environment.

```

2549 \dim_new:N \l__scripture_hanging_parindent_saved_dim

```

(End of definition for `\l__scripture_hanging_parindent_saved_dim`.)

`\l__scripture_hanging_chap_sep_dim` Calculated chapter separation used in the **hanging** environment.

```

2550 \dim_new:N \l__scripture_hanging_chap_sep_dim

```

(End of definition for `\l__scripture_hanging_chap_sep_dim`.)

`_scripture_hanging_ch:n` Position a chapter in a **hanging** environment.

```

2551 \cs_new_protected:Nn \_scripture_hanging_ch:n
2552 {
2553   \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
2554   \bool_if:NTF \l__scripture_chap_show_bool
2555     {
2556       \bool_if:NTF \l__scripture_chap_drop_bool
2557         { \_scripture_hanging_chap:n {#1} }
2558         {
2559           \bool_if:NTF \l__scripture_chap_para_bool

```



```

2560         { \__scripture_para_chap:n {#1} }
2561         { \__scripture_nodrop_chap:n {#1} }
2562     }
2563 }
2564 {
2565     \bool_if:NTF \l__scripture_chap_para_bool
2566     { \__scripture_para_chap:n {#1} }
2567     { \__scripture_nodrop_chap:n {#1} }
2568 }
2569 }

```

(End of definition for __scripture_hanging_ch:n.)

__scripture_hanging_chap:n Position a drop chapter in a **hanging** environment.

```

2570 \cs_new_protected:Nn \__scripture_hanging_chap:n
2571 {
2572     \dim_set:Nn \l__scripture_hanging_chap_sep_dim { \l__scripture_chap_sep_tl }
2573     \__scripture_drop_chap_set_up:n {#1}
2574     \dim_compare:nNnTF
2575     { \g__scripture_chap_width_dim + \l__scripture_hanging_chap_sep_dim }
2576     <
2577     { \leftmargin }
2578     {
2579         \bool_if:NT \l__scripture_hanging_chapter_indent_bool
2580         {
2581             \dim_set:Nn \l__scripture_hanging_chap_sep_dim
2582             { \leftmargin - \g__scripture_chap_width_dim }
2583         }
2584     }
2585     {
2586         \parshape 3 ~
2587         \dim_eval:n
2588         {
2589             \@totalleftmargin - \leftmargin + \g__scripture_chap_width_dim +
2590             \l__scripture_hanging_chap_sep_dim
2591         } ~
2592         \dim_eval:n
2593         {
2594             \linewidth + \leftmargin - \g__scripture_chap_width_dim -
2595             \l__scripture_hanging_chap_sep_dim
2596         } ~
2597         \dim_eval:n
2598         {
2599             \@totalleftmargin - \leftmargin + \g__scripture_chap_width_dim +
2600             \l__scripture_hanging_chap_sep_dim
2601         } ~
2602         \dim_eval:n
2603         {
2604             \linewidth + \leftmargin - \g__scripture_chap_width_dim -
2605             \l__scripture_hanging_chap_sep_dim
2606         } ~
2607         \@totalleftmargin ~ \linewidth
2608     }
2609     \par

```

```

2610 \bool_if:NT \l__scripture_chap_show_bool
2611 {
2612   \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
2613   {
2614     \addvspace
2615     {
2616       \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
2617     }
2618   }
2619 }
2620 \bool_if:NTF \l__scripture_hanging_chapter_indent_bool
2621 {
2622   \dim_set:Nn \l__scripture_hanging_chap_sep_dim
2623   {
2624     \l__scripture_hanging_chap_sep_dim +
2625     \l__scripture_hanging_parindent_saved_dim
2626   }
2627   \__scripture_nohang:
2628 }
2629 {
2630   \mode_leave_vertical:
2631   \dim_compare:nNnTF
2632   { \g__scripture_chap_width_dim + \l__scripture_hanging_chap_sep_dim }
2633   >
2634   { \l__scripture_hanging_leftmargin_tl }
2635   {
2636     \dim_compare:nNnTF
2637     {
2638       \g__scripture_chap_width_dim +
2639       \l__scripture_hanging_chap_sep_dim
2640     }
2641     >
2642     { \leftmargin }
2643     {
2644       \skip_horizontal:n { \l__scripture_hanging_hang_tl }
2645     }
2646     {
2647       \skip_horizontal:n
2648       {
2649         \l__scripture_hanging_chap_sep_dim +
2650         \g__scripture_chap_width_dim -
2651         \l__scripture_hanging_leftmargin_tl
2652       }
2653     }
2654   }
2655   {
2656     \dim_set:Nn \l__scripture_hanging_chap_sep_dim
2657     {
2658       \l__scripture_hanging_leftmargin_tl -
2659       \g__scripture_chap_width_dim
2660     }
2661   }
2662 }
2663 \hbox_set:Nn \l__scripture_chap_box

```

```

2664 {
2665     \box_move_down:nn
2666     { \baselineskip }
2667     {
2668         \hbox_overlap_left:n
2669         {
2670             \box_use:N \l__scripture_chap_tmp_box
2671             \skip_horizontal:N \l__scripture_hanging_chap_sep_dim
2672         }
2673     }
2674 }
2675 \box_set_ht:Nn \l__scripture_chap_box { \c_zero_dim }
2676 \box_set_dp:Nn \l__scripture_chap_box { \c_zero_dim }
2677 \group_begin:
2678 \hook_use:n { scripture / chap / before }
2679 \box_use:N \l__scripture_chap_box
2680 \hook_use:n { scripture / chap / after }
2681 \group_end:
2682 \bool_if:NTF \l__scripture_chap_show_verse_bool
2683 {
2684     \__scripture_verse_output:n { 1 }
2685 }
2686 { \__scripture_insert_verse_mark: }
2687 \__scripture_pilcrow_output:
2688 \bool_if:NT \l__scripture_verse_para_bool
2689 {
2690     \hook_gput_next_code:nn { para / after }
2691     {
2692         \int_gset_eq:NN \g__scripture_chap_par_prevgraf_int \prevgraf
2693         \int_compare:nNnT
2694             { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2695             { \penalty 10000 }
2696         \hook_gput_next_code:nn { para / before }
2697         {
2698             \int_set_eq:NN \clubpenalty \l__scripture_clubpenalty_saved_int
2699         }
2700     }
2701 }
2702 \hook_gput_next_code:nn { para / begin }
2703 {
2704     \int_compare:nNnTF
2705         { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2706         {
2707             \dim_compare:nNnTF
2708                 { \l__scripture_hanging_leftmargin_tl }
2709                 <
2710                 {
2711                     \g__scripture_chap_width_dim +
2712                     \l__scripture_hanging_chap_sep_dim
2713                 }
2714                 {
2715                     \parshape 2 ~
2716                     \dim_eval:n
2717                     {

```

```

2718         \@totalleftmargin - \leftmargin +
2719         \g__scripture_chap_width_dim +
2720         \l__scripture_hanging_chap_sep_dim
2721     } ~
2722     \dim_eval:n
2723     {
2724         \linewidth + \leftmargin - \g__scripture_chap_width_dim -
2725         \l__scripture_hanging_chap_sep_dim
2726     } ~
2727     \@totalleftmargin ~ \linewidth
2728 }
2729 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2730 \hook_gput_next_code:nn { para / begin }
2731 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2732 }
2733 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2734 \int_gzero:N \g__scripture_chap_par_prevgraf_int
2735 }
2736 }

```

(End of definition for _scripture_hanging_chap:n.)

_scripture_hanging_versen Position a \vs in a hanging environment. The verse is set to overlap to the left if at the beginning of a hanging paragraph.

```

2737 \cs_new_protected:Nn \_scripture_hanging_versen
2738 {
2739     \mode_if_vertical:TF
2740     {
2741         \tl_set_eq:NN \l_tmpa_tl \l__scripture_versesep_tl
2742         \tl_set_eq:NN
2743         \l__scripture_versesep_tl \l__scripture_hanging_versesep_tl
2744         \bool_lazy_and:nnTF
2745         { \legacy_if_p:n { @newlist } }
2746         { \g__scripture_suppress_next_pilcrow_bool }
2747         {
2748             \strut
2749             \bool_if:NTF \l__scripture_hanging_versesep_right_bool
2750             {
2751                 \hbox_overlap_left:n
2752                 {
2753                     \_scripture_versesep_output:n {#1}
2754                 }
2755             }
2756             {
2757                 \hbox_overlap_left:n
2758                 {
2759                     \hbox_overlap_right:n
2760                     {
2761                         \_scripture_versesep_output:n {#1}
2762                     }
2763                     \skip_horizontal:N \l__scripture_hanging_leftmargin_tl
2764                 }
2765             }
2766         }
2767     }
2768 }

```

```

2767     {
2768         \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2769         \strut
2770         \bool_if:NTF \l__scripture_hanging_verse_right_bool
2771         {
2772             \hbox_overlap_left:n
2773             {
2774                 \__scripture_verse_output:n {#1}
2775             }
2776         }
2777         {
2778             \hbox_overlap_left:n
2779             {
2780                 \hbox_overlap_right:n
2781                 {
2782                     \__scripture_verse_output:n {#1}
2783                 }
2784                 \skip_horizontal:N \l__scripture_hanging_leftmargin_tl
2785             }
2786         }
2787         \__scripture_pilcrow_output:
2788     }
2789     \__scripture_insert_verse_mark:
2790     \tl_set_eq:NN \l__scripture_verse_sep_tl \l_tmpa_tl
2791 }
2792 { \__scripture_verse_output:n {#1} }
2793 }

```

(End of definition for __scripture_hanging_verse:n.)

__scripture_nohang: Don't hang a paragraph in a **hanging** environment. Instead the paragraph is indented by \parindent from its normal hanging position.

```

2794 \cs_new_protected_nopar:Nn \__scripture_nohang:
2795 {
2796     \legacy_if:nTF { @newlist }
2797     { \skip_horizontal:n { \l__scripture_hanging_hang_tl } }
2798     { \__scripture_noindent_saved: }
2799     \hbox_to_wd:nn { \l__scripture_hanging_parindent_saved_dim } { }
2800     \peek_analysis_map_inline:n
2801     {
2802         \bool_lazy_or:nnT
2803         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
2804         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
2805         {
2806             \bool_gset_true:N \g__scripture_suppress_next_verse_para_bool
2807             \__scripture_set_pilcrow_hook:
2808         }
2809         \peek_analysis_map_break:n {##1}
2810     }
2811 }

```

(End of definition for __scripture_nohang:.)

__scripture_hanging_begin:n Begin a hanging section in a scripture quotation.

```

2812 \cs_new_protected:Nn \__scripture_hanging_begin:n
2813 {
2814   \bool_if:NT \l__scripture_active_inner_bool
2815   {
2816     \msg_error:nn { scripture } { nested-environment }
2817   }
2818   \bool_set_true:N \l__scripture_active_inner_bool
2819   \str_set:Nn \l__scripture_currenvir_str { hanging }
2820   \tl_if_novalue:nF {#1}
2821   {
2822     \keys_set:nn
2823     { scripture / hanging }
2824     {#1}
2825   }
2826   \DeclareDocumentCommand { \ch } { s o m }
2827   {
2828     \tl_if_novalue:nF {##2}
2829     {
2830       \tl_set:Nn \l__scripture_X_char_tl {##2}
2831     }
2832     \__scripture_hanging_ch:n {##3}
2833   }
2834   \bool_if:NTF \l__scripture_verse_para_bool
2835   { \cs_set_eq:NN \vs \__scripture_hanging_verse_para:n }
2836   { \cs_set_eq:NN \vs \__scripture_hanging_verse:n }
2837   \cs_set_eq:NN \nohang \__scripture_nohang:
2838   \dim_set_eq:NN \l__scripture_hanging_parindent_saved_dim \parindent
2839   \legacy_if:nTF { @newlist }
2840   {
2841     \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
2842   }
2843   { \dim_zero:N \l__scripture_outer_itemindent_dim }
2844   \cs_set_eq:NN \par \para_end:
2845   \hook_gc_clear_next_code:n { para / before }
2846   \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
2847   \list { }
2848   {
2849     \bool_lazy_or:nnF
2850     { \mode_if_vertical_p: }
2851     { \g__scripture_para_mode_vertical_bool }
2852     { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2853     \dim_zero:N \labelsep
2854     \dim_set:Nn \leftmargin
2855     {
2856       \l__scripture_hanging_leftmargin_tl +
2857       \l__scripture_hanging_hang_tl
2858     }
2859     \dim_set:Nn \rightmargin { \l__scripture_hanging_rightmargin_tl }
2860     \dim_set:Nn \itemindent
2861     {
2862       - \l__scripture_hanging_hang_tl -
2863       \l__scripture_outer_itemindent_dim
2864     }
2865     \dim_set:Nn \listparindent { - \l__scripture_hanging_hang_tl }

```

```

2866 \skip_set_eq:NN \parsep \parskip
2867 \skip_zero:N \partopsep
2868 \dim_compare:nNnTF
2869 { \parskip } > { \l__scripture_hanging_aboveskip_tl }
2870 { \skip_zero:N \topsep }
2871 {
2872 \skip_set:Nn \topsep
2873 {
2874 \__scripture_skip_diff:NN
2875 \l__scripture_hanging_aboveskip_tl \parskip
2876 }
2877 }
2878 \__scripture_setup_list_noindent:n { - \l__scripture_hanging_hang_tl }
2879 }
2880 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2881 { \addvspace { \baselineskip } }
2882 \int_gzero:N \g__scripture_chap_par_prevgraf_int
2883 \item
2884 \relax
2885 }

```

(End of definition for __scripture_hanging_begin:n.)

__scripture_hanging_end: End a hanging section in a scripture quotation.

```

2886 \cs_new_protected:Nn \__scripture_hanging_end:
2887 {
2888 \__scripture_calc_final_line_length:
2889 \dim_compare:nNnTF
2890 { \@outerparskip } > { \l__scripture_hanging_belowskip_tl }
2891 { \skip_zero:N \@topsepadd }
2892 {
2893 \skip_set:Nn \@topsepadd
2894 {
2895 \__scripture_skip_diff:NN
2896 \l__scripture_hanging_belowskip_tl \@outerparskip
2897 }
2898 }
2899 \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
2900 \legacy_if_set_false:n { @nparlist }
2901 \endlist
2902 \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2903 \hook_gput_next_code:nn { para / begin }
2904 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2905 }

```

(End of definition for __scripture_hanging_end:.)

3.6.4 The **midparachap** environment

__scripture_vbox_set_top:Nw Define missing \vbox_set_top:Nw function.

```

2906 \cs_new_protected:Npn \__scripture_vbox_set_top:Nw #1
2907 {
2908 \setbox #1 \vtop
2909 \c_group_begin_token

```

```

2910         \color_group_begin:
2911     }

(End of definition for \__scripture_vbox_set_top:Nw.)

\l_scripture_mid_para_chap_bool Flag if setting a paragraph containing a mid-paragraph chapter.
2912 \bool_new:N \l__scripture_mid_para_chap_bool

(End of definition for \l__scripture_mid_para_chap_bool.)

\l_scripture_mid_para_chap_line_box Store last line before a mid-paragraph chapter.
2913 \box_new:N \l__scripture_mid_para_chap_line_box

(End of definition for \l__scripture_mid_para_chap_line_box.)

\g_scripture_mid_para_chap_prevdepth_dim Store the \prevdepth of the final paragraph in a midparachap environment.
2914 \dim_new:N \g__scripture_mid_para_chap_prevdepth_dim

(End of definition for \g__scripture_mid_para_chap_prevdepth_dim.)

scripture_mid_para_chap_totalleftmargin_dim Save the \@totalleftmargin before starting a midparachap environment.
2915 \dim_new:N \l__scripture_mid_para_chap_totalleftmargin_dim

(End of definition for \l__scripture_mid_para_chap_totalleftmargin_dim.)

\g_scripture_mid_para_chap_prevgraf_int Store the \prevgraf of the final paragraph in a midparachap environment.
2916 \int_new:N \g__scripture_mid_para_chap_prevgraf_int

(End of definition for \g__scripture_mid_para_chap_prevgraf_int.)

\l_scripture_mid_para_chap_snap_box Store unpacked last line before a mid-paragraph chapter.
2917 \box_new:N \l__scripture_mid_para_chap_snap_box

(End of definition for \l__scripture_mid_para_chap_snap_box.)

\l_scripture_mid_para_chap_spacefactor_int Store space factor before a mid-paragraph chapter.
2918 \int_new:N \l__scripture_mid_para_chap_spacefactor_int

(End of definition for \l__scripture_mid_para_chap_spacefactor_int.)

\__scripture_mid_para_chap_end_orig:n Custom \end macro for the midparachap environment. This is only used when the
\__scripture_mid_para_chap_end:n verse/para and/or compact options are in use.
2919 \cs_set_eq:Nc \__scripture_mid_para_chap_end_orig:n { end ~ }
2920 \cs_new_protected_nopar:Nn \__scripture_mid_para_chap_end:n
2921 {
2922     \__scripture_mid_para_chap_end_orig:n {#1}
2923     \bool_if:NT \l__scripture_compact_bool
2924     { \c_space_token }
2925     \peek_remove_spaces:n
2926     {
2927         \peek_analysis_map_inline:n
2928         {
2929             \bool_case:n
2930             {
2931                 { \exp_args:No \token_if_letter_p:N {##1} }
2932                 { \__scripture_pilcrow_output: }

```



```

2933         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
2934         { \__scripture_set_pilcrow_hook: }
2935         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
2936         { \__scripture_set_pilcrow_hook: }
2937     }
2938     \peek_analysis_map_break:n {##1}
2939 }
2940 }
2941 }

```

(End of definition for __scripture_mid_para_chap_end_orig:n and __scripture_mid_para_chap_end:n.)

\l_scripture_mid_para_chap_box

Boxes containing a paragraph with a chapter break and its split if needed.

\l_scripture_mid_para_chap_split_box

```

2942 \box_new:N \l__scripture_mid_para_chap_box

```

```

2943 \box_new:N \l__scripture_mid_para_chap_split_box

```

```

2944 \box_new:N \l__scripture_mid_para_chap_split_top_box

```

(End of definition for \l_scripture_mid_para_chap_box, \l_scripture_mid_para_chap_split_box, and \l_scripture_mid_para_chap_split_top_box.)

_scripture_mid_para_drop_chap_output:

Output a mid-paragraph drop chapter.

```

2945 \cs_new_protected:Nn \__scripture_mid_para_drop_chap_output:
2946 {
2947     \unskip
2948     \int_set_eq:NN \l__scripture_mid_para_chap_spacefactor_int \spacefactor
2949     \strut
2950     \par
2951     \box_set_to_last:N \l__scripture_mid_para_chap_line_box
2952     \hbox_set:Nn \l__scripture_mid_para_chap_snap_box
2953     {
2954         \hbox_unpack_drop:N \l__scripture_mid_para_chap_line_box
2955         \unskip
2956         \unskip
2957         \unpenalty
2958     }
2959     \nointerlineskip
2960     \dim_compare:nNnTF
2961     {
2962         \box_wd:N \l__scripture_mid_para_chap_snap_box +
2963         \g__scripture_chap_width_dim +
2964         \l__scripture_chap_sep_tl
2965     }
2966     <
2967     { \linewidth - \leftskip - \rightskip - 3em }
2968     {
2969         \__scripture_drop_chap_output:
2970         \hbox_unpack_drop:N \l__scripture_mid_para_chap_snap_box
2971         \int_set_eq:NN \spacefactor \l__scripture_mid_para_chap_spacefactor_int
2972         \c_space_tl
2973         \__scripture_verse_output:n { 1 }
2974     }
2975     {
2976         \group_begin:
2977         \noindent

```

```

2978     \skip_horizontal:n { - \leftskip }
2979     \hbox_unpack_drop:N \l__scripture_mid_para_chap_snap_box
2980     \skip_zero:N \parfillskip
2981     \par
2982     \group_end:
2983     \__scripture_drop_chap_output:
2984     \bool_if:NTF \l__scripture_midparachap_show_verse_bool
2985     {
2986         \__scripture_verse_output:n { 1 }
2987     }
2988     { \__scripture_insert_verse_mark: }
2989 }
2990 }

```

(End of definition for __scripture_mid_para_drop_chap_output:.)

__scripture_mid_para_chap_begin:n Begin a paragraph containing a mid-paragraph chapter.

```

2991 \cs_new_protected:Nn \__scripture_mid_para_chap_begin:n
2992 {
2993     \str_set:Nn \l__scripture_currenvir_str { midparachap }
2994     \tl_if_novalue:nF {#1}
2995     {
2996         \keys_set:nn
2997         { scripture / midparachap }
2998         {#1}
2999     }
3000     \bool_lazy_and:nnT
3001     { \l__scripture_chap_show_verse_bool }
3002     { \l__scripture_chap_drop_bool }
3003     {
3004         \bool_set_true:N \l__scripture_midparachap_show_verse_bool
3005         \bool_set_false:N \l__scripture_chap_show_verse_bool
3006     }
3007     \par
3008     \bool_set_true:N \l__scripture_mid_para_chap_bool
3009     \__scripture_vbox_set_top:Nw \l__scripture_mid_para_chap_box
3010     \dim_zero:N \leftskip
3011     \dim_zero:N \rightskip
3012     \dim_set_eq:NN
3013     \l__scripture_mid_para_chap_totalleftmargin_dim \@totalleftmargin
3014     \dim_zero:N \@totalleftmargin
3015     \dim_set_eq:NN \hsize \linewidth
3016     \skip_zero:N \parskip
3017 }

```

(End of definition for __scripture_mid_para_chap_begin:n.)

__scripture_mid_para_chap_end: End a paragraph containing a mid-paragraph chapter.

```

3018 \cs_new_protected:Nn \__scripture_mid_para_chap_end:
3019 {
3020     \par
3021     \dim_gset_eq:NN \g__scripture_mid_para_chap_prevdepth_dim \prevdepth
3022     \box_set_to_last:N \l__scripture_mid_para_chap_line_box
3023     \hbox_set:Nn \l__scripture_mid_para_chap_snap_box
3024     {

```

```

3025     \hbox_unpack:N \l__scripture_mid_para_chap_line_box
3026     \unskip
3027     \unskip
3028     \unpenalty
3029   }
3030   \dim_gset:Nn \g__scripture_final_line_dim
3031   {
3032     \box_wd:N \l__scripture_mid_para_chap_snap_box +
3033     \l__scripture_mid_para_chap_totalleftmargin_dim
3034   }
3035   \box_clear:N \l__scripture_mid_para_chap_snap_box
3036   \nointerlineskip
3037   \int_compare:nNnT
3038     { \prevgraf } < { 3 }
3039   {
3040     \noindent
3041     \skip_horizontal:n
3042       { \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl }
3043     \dim_gadd:Nn \g__scripture_final_line_dim
3044       { \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl }
3045   }
3046   \box_use_drop:N \l__scripture_mid_para_chap_line_box
3047   \vbox_set_end:

```

Attempt to allow the mid-paragraph chapter box to break across pages.

```

3048   \mode_if_inner:TF
3049   {
3050     \noindent
3051     \box_use_drop:N \l__scripture_mid_para_chap_box
3052   }
3053   {
3054     \dim_compare:nNnTF
3055       { \box_ht_plus_dp:N \l__scripture_mid_para_chap_box }
3056       >
3057       { \pagegoal - \pagetotal - \baselineskip }
3058     {
3059       \int_set:Nn \vbadness { 10000 }
3060       \vbox_set_split_to_ht:NNn
3061         \l__scripture_mid_para_chap_split_box
3062         \l__scripture_mid_para_chap_box
3063       { \pagegoal - \pagetotal - \baselineskip }
3064       \vbox_set_top:Nn \l__scripture_mid_para_chap_split_top_box
3065       { \vbox_unpack_drop:N \l__scripture_mid_para_chap_split_box }
3066       \mark_get_marks_for_reinsertion:nNN
3067       { \box_use:N \l__scripture_mid_para_chap_split_top_box }
3068       \l_tmpa_tl \l_tmpb_tl
3069       \noindent
3070       \box_use_drop:N \l__scripture_mid_para_chap_split_top_box
3071       \tl_use:N \l_tmpa_tl
3072       \tl_use:N \l_tmpb_tl
3073       \goodbreak
3074       \mark_get_marks_for_reinsertion:nNN
3075       { \box_use:N \l__scripture_mid_para_chap_box }
3076       \l_tmpa_tl \l_tmpb_tl
3077       \box_use_drop:N \l__scripture_mid_para_chap_box

```

```

3078         \tl_use:N \l_tmpa_tl
3079         \tl_use:N \l_tmpb_tl
3080     }
3081     {
3082         \noindent
3083         \box_use_drop:N \l__scripture_mid_para_chap_box
3084     }
3085 }
3086 \par
3087 \dim_set_eq:NN \prevdepth \g__scripture_mid_para_chap_prevdepth_dim
3088 \int_zero:N \prevgraf
3089 \skip_vertical:n { - \parskip - \baselineskip }
3090 \noindent
3091 \skip_horizontal:N \g__scripture_final_line_dim
3092 \int_gset_eq:NN
3093     \g__scripture_chap_par_prevgraf_int
3094     \g__scripture_mid_para_chap_prevgraf_int
3095 \int_gset:Nn \g__scripture_current_group_level_a_int
3096     { \currentgrouplevel - \c_one_int }
3097 \__scripture_reset_par_at_group_level:
3098 }

```

(End of definition for __scripture_mid_para_chap_end:.)

3.6.5 The **narrow** environment

__scripture_narrow_begin:n Begin a narrow section in a scripture quotation.

```

3099 \cs_new_protected:Nn \__scripture_narrow_begin:n
3100 {
3101     \bool_if:NT \l__scripture_active_inner_bool
3102     {
3103         \msg_error:nn { scripture } { nested-environment }
3104     }
3105     \bool_set_true:N \l__scripture_active_inner_bool
3106     \str_set:Nn \l__scripture_currenvir_str { narrow }
3107     \tl_if_novalue:nF {#1}
3108     {
3109         \keys_set:nn
3110             { scripture / narrow }
3111             {#1}
3112     }
3113     \legacy_if:nTF { @newlist }
3114     {
3115         \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
3116     }
3117     { \dim_zero:N \l__scripture_outer_itemindent_dim }
3118     \cs_set_eq:NN \par \para_end:
3119     \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
3120     \list { }
3121     {
3122         \dim_zero:N \labelsep
3123         \dim_set:Nn \leftmargin { \l__scripture_narrow_leftmargin_tl }
3124         \dim_set:Nn \rightmargin { \l__scripture_narrow_rightmargin_tl }
3125         \dim_set_eq:NN \listparindent \parindent

```

```

3126 \bool_lazy_or:nnTF
3127 { \mode_if_vertical_p: }
3128 { \g__scripture_para_mode_vertical_bool }
3129 {
3130   \dim_set:Nn \itemindent
3131     { \parindent - \l__scripture_outer_itemindent_dim }
3132   \__scripture_setup_list_noindent:n { \parindent }
3133 }
3134 {
3135   \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3136   \dim_set:Nn \itemindent { - \l__scripture_outer_itemindent_dim }
3137   \__scripture_setup_list_noindent:n { \c_zero_dim }
3138 }
3139 \skip_set_eq:NN \parsep \parskip
3140 \skip_zero:N \partopsep
3141 \dim_compare:nNnTF
3142 { \parskip } > { \l__scripture_narrow_aboveskip_tl }
3143 { \skip_zero:N \topsep }
3144 {
3145   \skip_set:Nn \topsep
3146   {
3147     \__scripture_skip_diff:NN
3148     \l__scripture_narrow_aboveskip_tl \parskip
3149   }
3150 }
3151 }
3152 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3153 { \addvspace { \baselineskip } }
3154 \int_gzero:N \g__scripture_chap_par_prevgraf_int
3155 \bool_if:NT \l__scripture_verse_para_bool
3156 { \cs_set_eq:NN \par \__scripture_para_par: }
3157 \item
3158 \relax
3159 }

```

(End of definition for __scripture_narrow_begin:n.)

__scripture_narrow_end: End a narrow section in a scripture quotation.

```

3160 \cs_new_protected:Nn \__scripture_narrow_end:
3161 {
3162   \cs_set_eq:NN \par \para_end:
3163   \__scripture_calc_final_line_length:
3164   \dim_compare:nNnTF
3165   { \@outerparskip } > { \l__scripture_narrow_belowskip_tl }
3166   { \skip_zero:N \@topsepadd }
3167   {
3168     \skip_set:Nn \@topsepadd
3169     {
3170       \__scripture_skip_diff:NN
3171       \l__scripture_narrow_belowskip_tl \@outerparskip
3172     }
3173   }
3174   \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
3175   \legacy_if_set_false:n { @nparlist }

```

```

3176 \endlist
3177 \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3178 \hook_gput_next_code:nn { para / begin }
3179 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
3180 }

```

(End of definition for __scripture_narrow_end:.)

3.6.6 The **poetry** environment

scripture/poetry/para/after This hook is an internal hook used to adjust the \parshape within a **poetry** environment after drop chapter numbers.

```

3181 \hook_new:n { scripture / poetry / para / after }

```

(End of definition for scripture/poetry/para/after. This variable is documented on page ??.)

\g__scripture_poetry_end_bool Flag indicating that a **poetry** environment just ended.

```

3182 \bool_new:N \g__scripture_poetry_end_bool

```

(End of definition for \g__scripture_poetry_end_bool.)

scripture_poetry_midparachap_show_verse_bool Flag indicating whether the verse number should be printed immediately after a \ch or not in a **poetry** environment.

```

3183 \bool_new:N \l__scripture_poetry_midparachap_show_verse_bool

```

(End of definition for \l__scripture_poetry_midparachap_show_verse_bool.)

\l__scripture_poetry_chap_indent_dim Store calculated indent and sep when a drop chapter appears in a **poetry** environment.

```

\l__scripture_poetry_chap_sep_dim
3184 \dim_new:N \l__scripture_poetry_chap_indent_dim

```

```

3185 \dim_new:N \l__scripture_poetry_chap_sep_dim

```

(End of definition for \l__scripture_poetry_chap_indent_dim and \l__scripture_poetry_chap_sep_dim.)

scripture_poetry_chap_parshape_correction_dim Correction to \parshape in chapter affected lines in a **poetry** environment.

```

3186 \dim_new:N \l__scripture_poetry_chap_parshape_correction_dim

```

(End of definition for \l__scripture_poetry_chap_parshape_correction_dim.)

\l__scripture_poetry_prevgraf_int Store the number of lines a line of poetry takes up.

```

3187 \int_new:N \l__scripture_poetry_prevgraf_int

```

(End of definition for \l__scripture_poetry_prevgraf_int.)

_scripture_poetry_text_right:nn Align a word to the right and finish a paragraph, pushing to the next line if there is not space for the word.

```

3188 \cs_new_protected:Nn \_scripture_poetry_text_right:nn
3189 {
3190   {
3191     \unskip
3192     \hfil
3193     \penalty 50
3194     \skip_horizontal:n {#1}
3195     \hbox:n {}
3196     \nobreak

```

```

3197     \hfill
3198     #2
3199     \skip_zero:N \parfillskip
3200     \int_zero:N \finalhyphendemerits
3201   }
3202 }

```

(End of definition for `__scripture_poetry_text_right:nn`.)

`__scripture_selah_output:` Output Selah at the end of a line.

```

3203 \cs_new_protected:Nn \__scripture_selah_output:
3204 {
3205   \group_begin:
3206   \l__scripture_selah_font_tl
3207   \__scripture_poetry_text_right:nn
3208   { \l__scripture_selah_sep_tl }
3209   { \__scripture_selah_format:n { \l__scripture_selah_text_tl } }
3210   \group_end:
3211 }

```

(End of definition for `__scripture_selah_output:.`)

`__scripture_poetry_ch:n` Position a chapter in a **poetry** environment.

```

3212 \cs_new_protected:Nn \__scripture_poetry_ch:n
3213 {
3214   \bool_if:NTF \l__scripture_chap_show_bool
3215   {
3216     \bool_if:NTF \l__scripture_chap_drop_bool
3217     { \__scripture_poetry_chap:n {#1} }
3218     {
3219       \bool_if:NTF \l__scripture_chap_para_bool
3220       { \__scripture_para_chap:n {#1} }
3221       { \__scripture_poetry_nodrop_chap:n {#1} }
3222     }
3223   }
3224   {
3225     \bool_if:NTF \l__scripture_chap_para_bool
3226     { \__scripture_para_chap:n {#1} }
3227     { \__scripture_poetry_nodrop_chap:n {#1} }
3228   }
3229 }

```

(End of definition for `__scripture_poetry_ch:n`.)

`__scripture_poetry_chap:n` Position a drop chapter in a **poetry** environment.

```

3230 \cs_new_protected:Nn \__scripture_poetry_chap:n
3231 {
3232   \__scripture_drop_chap_set_up:n {#1}
3233   \int_gset_eq:NN \g__scripture_current_group_level_c_int \currentgrouplevel
3234   \__scripture_zero_chap_par_prevgraf_at_group_level:
3235   \bool_if:NT \l__scripture_chap_show_bool
3236   {
3237     \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
3238     {
3239       \addvspace

```

```

3240         {
3241             \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
3242         }
3243     }
3244 }
3245 \cs_if_eq:NNTF \vs \__scripture_poetry_mode_vertical Verse:n
3246 {
3247     \dim_zero:N \l__scripture_poetry_chap_indent_dim
3248     \noindent
3249 }
3250 {
3251     \dim_set:Nn \l__scripture_poetry_chap_indent_dim
3252     { \l__scripture_poetry_indent_tl }
3253     \mode_leave_vertical:
3254 }
3255 \dim_set:Nn \l__scripture_poetry_chap_sep_dim { \l__scripture_chap_sep_tl }
3256 \dim_compare:nNnT
3257 { \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl }
3258 <
3259 {
3260     \l__scripture_poetry_leftmargin_tl +
3261     \l__scripture_poetry_chap_indent_dim
3262 }
3263 {
3264     \dim_set:Nn \l__scripture_poetry_chap_sep_dim
3265     {
3266         \l__scripture_poetry_leftmargin_tl +
3267         \l__scripture_poetry_chap_indent_dim -
3268         \g__scripture_chap_width_dim
3269     }
3270 }
3271 \hbox_set:Nn \l__scripture_chap_box
3272 {
3273     \box_move_down:nn
3274     { \baselineskip }
3275     {
3276         \hbox_overlap_left:n
3277         {
3278             \box_use:N \l__scripture_chap_tmp_box
3279             \skip_horizontal:N \l__scripture_poetry_chap_sep_dim
3280         }
3281     }
3282 }
3283 \box_set_ht:Nn \l__scripture_chap_box { \c_zero_dim }
3284 \box_set_dp:Nn \l__scripture_chap_box { \c_zero_dim }
3285 \dim_set:Nn \l__scripture_poetry_chap_parshape_correction_dim
3286 {
3287     \g__scripture_chap_width_dim +
3288     \l__scripture_poetry_chap_sep_dim -
3289     \l__scripture_poetry_leftmargin_tl -
3290     \l__scripture_poetry_chap_indent_dim
3291 }
3292 \parshape 2 ~
3293 \dim_eval:n

```



```

3294     {
3295         \@totalleftmargin + \l__scripture_poetry_chap_parshape_correction_dim
3296     } ~
3297     \dim_eval:n
3298     { \linewidth - \l__scripture_poetry_chap_parshape_correction_dim } ~
3299     \dim_eval:n
3300     { \@totalleftmargin + \l__scripture_poetry_big_indent_tl } ~
3301     \dim_eval:n
3302     { \linewidth - \l__scripture_poetry_big_indent_tl }

```

Set the \parshape for the second line after a drop chapter number.

```

3303     \dim_compare:nNnT
3304     { \g__scripture_chap_width_dim + \l__scripture_poetry_chap_sep_dim }
3305     >
3306     { \l__scripture_poetry_leftmargin_tl }
3307     {
3308         \hook_gput_next_code:nn { scripture / poetry / para / after }
3309         {
3310             \int_compare:nNnT
3311             { \l__scripture_poetry_prevgraf_int } = { \c_one_int }
3312             {
3313                 \cs_if_eq:NNTF \vs \__scripture_poetry_mode_vertical_verse:n

```

This \parshape is in effect when the next line is the first half of a poetry verse.

```

3314     {
3315         \dim_compare:nNnTF
3316         {
3317             \g__scripture_chap_width_dim +
3318             \l__scripture_chap_sep_tl
3319         }
3320         <
3321         { \l__scripture_poetry_leftmargin_tl }
3322         {
3323             \dim_zero:N
3324             \l__scripture_poetry_chap_parshape_correction_dim
3325         }
3326         {
3327             \dim_set:Nn
3328             \l__scripture_poetry_chap_parshape_correction_dim
3329             {
3330                 \g__scripture_chap_width_dim +
3331                 \l__scripture_chap_sep_tl -
3332                 \l__scripture_poetry_leftmargin_tl
3333             }
3334         }
3335         \int_compare:nNnT
3336         { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3337         {
3338             \parshape 2 ~
3339             \dim_eval:n
3340             {
3341                 \@totalleftmargin +
3342                 \l__scripture_poetry_chap_parshape_correction_dim
3343             } ~
3344             \dim_eval:n

```

```

3345         {
3346             \linewidth -
3347             \l__scripture_poetry_chap_parshape_correction_dim
3348         } ~
3349     \dim_eval:n
3350     {
3351         \@totalleftmargin +
3352         \l__scripture_poetry_big_indent_tl
3353     } ~
3354     \dim_eval:n
3355     {
3356         \linewidth -
3357         \l__scripture_poetry_big_indent_tl
3358     }
3359 }
3360 }

```

This `\parshape` is in effect when the next line is the second half of a poetry verse.

```

3361 {
3362     \dim_compare:nNnTF
3363     {
3364         \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl
3365     }
3366     <
3367     {
3368         \l__scripture_poetry_leftmargin_tl +
3369         \l__scripture_poetry_indent_tl
3370     }
3371     {
3372         \dim_zero:N
3373         \l__scripture_poetry_chap_parshape_correction_dim
3374     }
3375     {
3376         \dim_set:Nn
3377         \l__scripture_poetry_chap_parshape_correction_dim
3378         {
3379             \g__scripture_chap_width_dim +
3380             \l__scripture_chap_sep_tl -
3381             \l__scripture_poetry_leftmargin_tl -
3382             \l__scripture_poetry_indent_tl
3383         }
3384     }
3385     \parshape 2 ~
3386     \dim_eval:n
3387     {
3388         \@totalleftmargin +
3389         \l__scripture_poetry_chap_parshape_correction_dim
3390     } ~
3391     \dim_eval:n
3392     {
3393         \linewidth -
3394         \l__scripture_poetry_chap_parshape_correction_dim
3395     } ~
3396     \dim_eval:n
3397     {

```

```

3398             \@totalleftmargin + \l__scripture_poetry_big_indent_tl
3399         } ~
3400         \dim_eval:n
3401         { \linewidth - \l__scripture_poetry_big_indent_tl }
3402     }
3403 }
3404 }
3405 }
3406 \group_begin:
3407 \hook_use:n { scripture / chap / before }
3408 \box_use:N \l__scripture_chap_box
3409 \hook_use:n { scripture / chap / after }
3410 \group_end:
3411 \bool_lazy_or:nnTF
3412   { \l__scripture_chap_show_verse_bool }
3413   { \l__scripture_poetry_midparachap_show_verse_bool }
3414   { \__scripture_verse_output:n { 1 } }
3415   { \__scripture_insert_verse_mark: }
3416 \hook_use:n { scripture / poetry / pilcrow }
3417 }

```

(End of definition for __scripture_poetry_chap:n.)

__scripture_poetry_nodrop_chap:n Format a non-drop chapter for **poetry** environment in a **scripture** quotation.

```

3418 \cs_new_protected:Nn \__scripture_poetry_nodrop_chap:n
3419 {
3420   \tl_gset:Nn \scripturecurrentchapter {#1}
3421   \tl_gset:Nn \scripturecurrentverse { 1 }
3422   \group_begin:
3423   \bool_if:NT \l__scripture_chap_show_bool
3424   {
3425     \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
3426     {
3427       \addvspace
3428       {
3429         \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
3430       }
3431     }
3432   }
3433   \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical_verse:n
3434   { \noindent }
3435   \bool_if:NTF \l__scripture_chap_show_bool
3436   {
3437     \hbox_overlap_left:n
3438     {
3439       \hook_use:n { scripture / chap / before }
3440       \group_begin:
3441       \l__scripture_chap_font_tl
3442       \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
3443       { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
3444       \__scripture_select_language:n { chapter }
3445       \__scripture_chap_format:n {#1}
3446       \group_end:
3447       \skip_horizontal:N \l__scripture_chap_sep_tl

```

```

3448         \hook_use:n { scripture / chap / after }
3449         \bool_lazy_or:nnTF
3450             { \l__scripture_chap_show_verse_bool }
3451             { \l__scripture_poetry_midparachap_show_verse_bool }
3452             { \__scripture_verse_output:n { 1 } }
3453             { \__scripture_insert_verse_mark: }
3454     }
3455 }
3456 {
3457     \hook_use:n { scripture / chap / before }
3458     \hook_use:n { scripture / chap / after }
3459     \vs { 1 }
3460 }
3461 \group_end:
3462 \hook_use:n { scripture / poetry / pilcrow }
3463 }

```

(End of definition for __scripture_poetry_nodrop_chap:n.)

__scripture_poetry_mode_vertical_verse:n Format a verse for poetry scripture quotations in vertical mode.

```

3464 \cs_new_protected:Nn \__scripture_poetry_mode_vertical_verse:n
3465 {
3466     \group_begin:
3467     \tl_set_eq:NN \l__scripture_verse_sep_tl \l__scripture_poetry_verse_sep_tl
3468     \hook_gclear_next_code:n { scripture / pilcrow }
3469     \strut
3470     \int_compare:nNnTF { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3471     {
3472         \__scripture_verse_output:n {#1}
3473     }
3474     {
3475         \bool_if:NTF \l__scripture_poetry_verse_right_bool
3476         {
3477             \hbox_overlap_left:n
3478             {
3479                 \__scripture_verse_output:n {#1}
3480             }
3481         }
3482         {
3483             \hbox_overlap_left:n
3484             {
3485                 \hbox_overlap_right:n
3486                 {
3487                     \__scripture_verse_output:n {#1}
3488                 }
3489                 \skip_horizontal:N \l__scripture_poetry_leftmargin_tl
3490             }
3491         }
3492         \__scripture_insert_verse_mark:
3493     }
3494     \group_end:
3495     \bool_if:NF \g__scripture_suppress_next_pilcrow_bool
3496     { \hook_use:n { scripture / poetry / pilcrow } }
3497     \cs_set_eq:NN \vs \__scripture_verse_output:n
3498 }

```

(End of definition for `__scripture_poetry_mode_vertical_verse:n`.)

`__scripture_poetry_mode_horizontal_verse:n` Format a verse for poetry scripture quotations in horizontal mode.

```

3499 \cs_new_protected:Nn \__scripture_poetry_mode_horizontal_verse:n
3500 {
3501   \group_begin:
3502   \tl_set_eq:NN \l__scripture_verse_sep_tl \l__scripture_poetry_verse_sep_tl
3503   \int_compare:nNnTF { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3504   {
3505     \__scripture_poetry_mode_vertical_verse:n {#1}
3506   }
3507   {
3508     \skip_horizontal:n { - \listparindent }
3509     \__scripture_poetry_mode_vertical_verse:n {#1}
3510     \skip_horizontal:N \listparindent
3511   }
3512   \group_end:
3513   \cs_set_eq:NN \vs \__scripture_verse_output:n
3514 }

```

(End of definition for `__scripture_poetry_mode_horizontal_verse:n`.)

`__scripture_poetry_par:` par macro for poetry sections

```

3515 \cs_new_protected:Nn \__scripture_poetry_par:
3516 {
3517   \para_end:
3518   \int_set:Nn \l__scripture_poetry_prevgraf_int { \int_use:N \prevgraf }
3519   \parshape 2 ~
3520   \@totalleftmargin ~
3521   \linewidth ~
3522   \dim_eval:n
3523   { \@totalleftmargin + \l__scripture_poetry_big_indent_tl } ~
3524   \dim_eval:n
3525   { \linewidth - \l__scripture_poetry_big_indent_tl }
3526   \hook_use:n { scripture / poetry / para / after }
3527 }

```

(End of definition for `__scripture_poetry_par:.`)

`__scripture_obeylines:` Special version of obeylines which doesn't insert par at end of environment.

```

3528 \group_begin:
3529 \char_set_catcode_active:N \^M
3530 \cs_gset_protected_nopar:Nn \__scripture_obeylines:
3531 {
3532   \char_set_catcode_active:N \^M
3533   \cs_set_eq:NN \^M \__scripture_obeylines_eol:
3534 }
3535 \group_end:
3536 \cs_new_protected:Nn \__scripture_obeylines_eol:
3537 {
3538   \mode_if_vertical:TF
3539   {
3540     \cs_set_eq:NN \vs \__scripture_poetry_mode_vertical_verse:n
3541     \hook_gput_next_code:nn { scripture / poetry / para / after }

```

```

3542         { \bool_set_false:N \l__scripture_poetry_midparachap_show_verse_bool }
3543     }
3544     { \cs_set_eq:NN \vs \__scripture_poetry_mode_horizontal_verse:n }
3545 \peek_analysis_map_inline:n
3546 {
3547     \bool_case:nF
3548     {
3549         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \ch }
3550         {
3551             \bool_set_eq:NN
3552             \l__scripture_poetry_midparachap_show_verse_bool
3553             \l__scripture_midparachap_show_verse_bool
3554             \__scripture_poetry_par:
3555             \peek_analysis_map_break:n {##1}
3556         }
3557         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newch }
3558         {
3559             \bool_set_eq:NN
3560             \l__scripture_poetry_midparachap_show_verse_bool
3561             \l__scripture_midparachap_show_verse_bool
3562             \__scripture_poetry_par:
3563             \peek_analysis_map_break:n {##1}
3564         }
3565         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \end }
3566         { \peek_analysis_map_break:n {##1} }
3567         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \extraskip }
3568         { \peek_analysis_map_break:n {##1} }
3569         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nopilcrow }
3570         {
3571             \__scripture_poetry_par:
3572             \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical_verse:n
3573             { \noindent }
3574             \peek_analysis_map_break:n
3575             {
3576                 \__scripture_fix_protrusion:o {##1}
3577             }
3578         }
3579         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
3580         {
3581             \__scripture_poetry_par:
3582             \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical_verse:n
3583             { \noindent }
3584             \peek_analysis_map_break:n {##1}
3585         }
3586         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
3587         {
3588             \__scripture_poetry_par:
3589             \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical_verse:n
3590             { \noindent }
3591             \peek_analysis_map_break:n {##1}
3592         }
3593     }
3594     {
3595         \exp_args:No
3596         \token_if_eq_meaning_p:NN {##1} \__scripture_obeylines_eol:

```

```

3596     }
3597     { \cs_set_eq:NN \vs \__scripture_poetry_mode_vertical_verse:n }
3598   }
3599   {
3600     \__scripture_poetry_par:
3601     \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical_verse:n
3602       { \noindent }
3603     \cs_set_eq:NN \vs \__scripture_verse_output:n
3604     \bool_if:NT \l__scripture_verse_para_bool
3605       {
3606         \__scripture_pilcrow_output:
3607         \bool_set_false:N \l__scripture_verse_para_bool
3608         \bool_set_false:N \l__scripture_pilcrow_bool
3609       }
3610     \peek_analysis_map_break:n
3611     {
3612       \__scripture_fix_protrusion:o {##1}
3613     }
3614   }
3615 }
3616 }

```

(End of definition for __scripture_obeylines:.)

__scripture_poetry_begin: Begin a poetry section in a scripture quotation.

```

3617 \cs_new_protected:Nn \__scripture_poetry_begin:
3618 {
3619   \bool_if:NT \l__scripture_active_inner_bool
3620     {
3621       \msg_error:nn { scripture } { nested-environment }
3622     }
3623   \bool_set_true:N \l__scripture_active_inner_bool
3624   \str_set:Nn \l__scripture_currenvir_str { poetry }
3625   \bool_gset_false:N \g__scripture_poetry_midparachap_show_verse_bool
3626   \DeclareDocumentCommand { \ch } { s o m }
3627     {
3628       \tl_if_novalue:nF {##2}
3629       {
3630         \tl_set:Nn \l__scripture_X_char_tl {##2}
3631       }
3632       \__scripture_poetry_ch:n {##3}
3633     }
3634   \cs_set_eq:NN \vs \__scripture_poetry_mode_vertical_verse:n
3635   \cs_set_eq:NN \selah \__scripture_selah_output:
3636   \cs_set_protected_nopar:Npn \textright ##1
3637     {
3638       \__scripture_poetry_text_right:nn
3639       { \l__scripture_text_right_sep_tl } {##1}
3640     }
3641   \legacy_if:nTF { @newlist }
3642     {
3643       \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
3644     }
3645     { \dim_zero:N \l__scripture_outer_itemindent_dim }

```

```

3646 \cs_set_eq:NN \para \para_end:
3647 \cs_set_eq:NN \noindent \__scripture_noindent_saved:
3648 \hook_gc_clear_next_code:n { para / before }
3649 \hook_gput_next_code:nn { scripture / poetry / pilcrow }
3650 {
3651   \__scripture_pilcrow_output:
3652   \bool_set_false:N \l__scripture_pilcrow_bool
3653   \bool_set_false:N \l__scripture_verse_para_bool
3654 }
3655 \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
3656 \list { }
3657 {
3658   \bool_lazy_or:nnF
3659   { \mode_if_vertical_p: }
3660   { \g__scripture_para_mode_vertical_bool }
3661   { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
3662   \dim_zero:N \labelsep
3663   \dim_set:Nn \leftmargin { \l__scripture_poetry_leftmargin_tl }
3664   \dim_set:Nn \rightmargin { \l__scripture_poetry_rightmargin_tl }
3665   \dim_set:Nn \itemindent
3666   {
3667     \l__scripture_poetry_indent_tl - \l__scripture_outer_itemindent_dim
3668   }
3669   \dim_set:Nn \listparindent { \l__scripture_poetry_indent_tl }
3670   \skip_zero:N \partopsep
3671   \skip_zero:N \partopsep
3672   \dim_compare:nNnTF { \parskip } > { \l__scripture_poetry_aboveskip_tl }
3673   { \skip_zero:N \topsep }
3674   {
3675     \skip_set:Nn \topsep
3676     {
3677       \__scripture_skip_diff:NN
3678       \l__scripture_poetry_aboveskip_tl \parskip
3679     }
3680   }
3681   \__scripture_setup_list_noindent:n { \l__scripture_poetry_indent_tl }
3682 }
3683 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3684 { \addvspace { \baselineskip } }
3685 \int_gzero:N \g__scripture_chap_par_prevgraf_int
3686 \raggedright
3687 \dim_set:Nn \parindent { \l__scripture_poetry_indent_tl }
3688 \item
3689 \relax
3690 }

```

(End of definition for __scripture_poetry_begin:.)

`__scripture_poetry_end:` End a poetry section in a scripture quotation.

```

3691 \cs_new_protected:Nn \__scripture_poetry_end:
3692 {
3693   \__scripture_calc_final_line_length:
3694   \dim_compare:nNnTF
3695   { \@outerparskip } > { \l__scripture_poetry_belowskip_tl }

```



```

3696     { \skip_zero:N \@topsepadd }
3697     {
3698         \skip_set:Nn \@topsepadd
3699         {
3700             \__scripture_skip_diff:NN
3701             \l__scripture_poetry_belowskip_tl \@outerparskip
3702         }
3703     }
3704     \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
3705     \legacy_if_set_false:n { @nparlist }
3706     \endlist
3707     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3708     \hook_gc_clear_next_code:n { scripture / poetry / pilcrow }
3709     \bool_gset_true:N \g__scripture_poetry_end_bool
3710     \hook_gput_next_code:nn { para / begin }
3711     {
3712         \bool_gset_false:N \g__scripture_poetry_end_bool
3713         \parshape 1 ~ \@totalleftmargin ~ \linewidth
3714     }
3715 }

```

(End of definition for `__scripture_poetry_end:`.)

3.7 Public L^AT_EX Interface

`\scripturecurrentbook` Globally stores the most recent argument of `\ch` and `\vs`. These could be used in hooks, for example. They are initialised from the `book`, `chapter` and `verse` options to the main `scripture` environment.

```

3716 \tl_new:N \scripturecurrentbook
3717 \tl_new:N \scripturecurrentchapter
3718 \tl_new:N \scripturecurrentverse

```

(End of definition for `\scripturecurrentbook`, `\scripturecurrentchapter`, and `\scripturecurrentverse`. These functions are documented on page ??.)

`\scripturesetup` #1 : `<options>` to set.
Set key-value settings.

```

3719 \NewDocumentCommand { \scripturesetup } { m }
3720 {
3721     \keys_set:nn { scripture } {#1}
3722 }

```

(End of definition for `\scripturesetup`. This function is documented on page 8.)

`\scripturestyle` #1 : `<name>` of the style.
#2 : `<options>` to set for the style.
Set key-value settings to be applied when using the `style` option with value `name`. You may want to pass `defaults` as the first option to ensure any unset options in your style are expected defaults. The starred version appends the `options` to any existing ones.

```

3723 \NewDocumentCommand { \scripturestyle } { s m m }
3724 {
3725     \__scripture_setup_style:nnn {#1} {#2} {#3}
3726 }

```

(End of definition for `\scripturestyle`. This function is documented on page 8.)

`\textscripture` #1 : \langle reference \rangle of quotation.

#2 : \langle options \rangle to set.

#3 : \langle quotation \rangle text.

Set an inline quotation. This is equivalent to calling the `scripture` environment with the `inline` option set.

```

3727 \NewDocumentCommand { \textscripture } { o o +m }
3728 {
3729   \tl_if_novalue:nTF {#2}
3730   {
3731     \begin { scripture } [ #1 ] [ inline ]
3732   }
3733   {
3734     \begin { scripture } [ #1 ] [ inline, #2 ]
3735   }
3736   #3
3737   \end { scripture }
3738 }
```

(End of definition for `\textscripture`. This function is documented on page 9.)

`scripture` #1 : \langle reference \rangle of quotation.

#2 : \langle options \rangle to set.

Format a scripture quotation with optional reference.

```

3739 \NewDocumentEnvironment { scripture } { o o }
3740 {
3741   \__scripture_begin:nn {#1} {#2}
```

#1 : \langle options \rangle to set.

```

3742 \DeclareDocumentEnvironment { center } { o }
3743 {
3744   \__scripture_center_begin:n {##1}
3745   \__scripture_reference_start_peek:
3746 }
3747 { \__scripture_center_end: }
```

#1 : \langle options \rangle to set.

```

3748 \DeclareDocumentEnvironment { flushright } { o }
3749 {
3750   \__scripture_flushright_begin:n {##1}
3751   \__scripture_reference_start_peek:
3752 }
3753 { \__scripture_flushright_end: }
```

#1 : \langle options \rangle to set.

```

3754 \DeclareDocumentEnvironment { hanging } { o }
3755 {
3756   \__scripture_hanging_begin:n {##1}
3757   \__scripture_reference_start_peek:
3758 }
3759 { \__scripture_hanging_end: }
```

```

midparachap 3760 \DeclareDocumentEnvironment { midparachap } { o }
3761 {
3762     \__scripture_mid_para_chap_begin:n {##1}
3763     \__scripture_reference_start_peek:
3764 }
3765 { \__scripture_mid_para_chap_end: }

#1 : <options> to set.

narrow 3766 \DeclareDocumentEnvironment { narrow } { o }
3767 {
3768     \__scripture_narrow_begin:n {##1}
3769     \__scripture_reference_start_peek:
3770 }
3771 { \__scripture_narrow_end: }

#1 : <options> to set.
poetry The poetryaux environment is required because \__scripture_obeylines: breaks
the reading of optional arguments.
3772 \DeclareDocumentEnvironment { poetryaux } { o }
3773 {
3774     \tl_if_novalue:nF {##1}
3775     {
3776         \keys_set:nn
3777             { scripture / poetry }
3778             {##1}
3779     }
3780     \__scripture_poetry_begin:
3781     \__scripture_reference_start_peek:
3782 }
3783 { \__scripture_poetry_end: }
3784 \DeclareDocumentEnvironment { poetry } { }
3785 {
3786     \__scripture_obeylines:
3787     \poetryaux
3788 }
3789 { \endpoetryaux }

\bool_if:NT \l__scripture_inline_bool
3790 {
3791     \tl_set:Nn \l__scripture_aboveskip_tl { Opt }
3792     \tl_set:Nn \l__scripture_belowskip_tl { Opt }
3793     \tl_set:Nn \l__scripture_leftmargin_tl { Opt }
3794     \tl_set:Nn \l__scripture_rightmargin_tl { Opt }
3795     \cs_set_eq:NN \__scripture_ref_format:n \__scripture_inline_ref_format:n
3796     \tl_set_eq:NN \l__scripture_ref_sep_tl \l__scripture_inline_ref_sep_tl
3797     \cs_set_eq:NN
3798         \__scripture_version_format:n \__scripture_inline_version_format:n
3799     \tl_set_eq:NN
3800         \l__scripture_version_delim_tl \l__scripture_inline_version_delim_tl
3801     \bool_set_false:N \l__scripture_chap_para_bool
3802     \bool_set_true:N \l__scripture_compact_bool
3803     \bool_set_false:N \l__scripture_heading_show_bool
3804

```

```

3805     }
3806 \bool_if:NT \l__scripture_chap_para_bool
3807 {
3808     \RenewDocumentEnvironment { midparachap } { o }
3809     { \ignorespaces }
3810     { \unskip }
3811 }
3812 \bool_if:NT \l__scripture_verse_para_bool
3813 {
3814     \cs_set_eq:NN \vs \__scripture_verse_para_output:n
3815     \cs_set_eq:NN \par \__scripture_para_par:
3816     \cs_set_eq:NN \@par \__scripture_para_par:
3817     \cs_set_eq:NN \@doendpe \__scripture_doendpe:
3818     \cs_set_eq:NN \noindent \__scripture_para_noindent:
3819     \cs_set_protected:Npn \\\
3820     {
3821         \mode_if_horizontal:TF
3822         { \unskip }
3823         { \@nolnerr }
3824     }
3825     \RenewDocumentEnvironment { midparachap } { o }
3826     {
3827         \str_set:Nn \l__scripture_currenvir_str { midparachap }
3828         \cs_set_eq:NN \noindent \__scripture_noindent_saved:
3829         \cs_set_eq:cN { end ~ } \__scripture_mid_para_chap_end:n
3830         \__scripture_reference_start_peek:
3831     }
3832     {
3833         \para_end:
3834         \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
3835     }
3836 }
3837 \bool_if:NT \l__scripture_compact_bool
3838 {
3839     \cs_set_eq:NN \noindent \__scripture_para_noindent:
3840     \bool_if:NF \l__scripture_verse_para_bool
3841     {
3842         \bool_set_false:N \l__scripture_chap_drop_bool
3843         \bool_set_false:N \l__scripture_chap_drop_llap_bool
3844         \bool_set_false:N \l__scripture_chap_drop_smash_bool
3845         \bool_set_false:N \l__scripture_chap_para_bool
3846         \cs_set_eq:NN \vs \__scripture_verse_compact_output:n
3847     }
3848     \bool_gset:Nn \g__scripture_mode_vertical_inner_bool
3849     { \mode_if_vertical_p: }
3850     \RenewDocumentEnvironment { center } { o }
3851     {
3852         \str_set:Nn \l__scripture_currenvir_str { center }
3853         \__scripture_reference_start_peek:
3854     }
3855     { \__scripture_compact_end_inner: }
3856     \RenewDocumentEnvironment { flushright } { o }
3857     {
3858         \str_set:Nn \l__scripture_currenvir_str { flushright }

```

```

3859     \__scripture_reference_start_peek:
3860   }
3861   { \__scripture_compact_end_inner: }
3862   \RenewDocumentEnvironment { hanging } { o }
3863   {
3864     \str_set:Nn \l__scripture_currenvir_str { hanging }
3865     \__scripture_reference_start_peek:
3866   }
3867   { \__scripture_compact_end_inner: }
3868   \RenewDocumentEnvironment { midparachap } { o }
3869   {
3870     \str_set:Nn \l__scripture_currenvir_str { midparachap }
3871     \cs_set_eq:cN { end ~ } \__scripture_mid_para_chap_end:n
3872     \__scripture_reference_start_peek:
3873   }
3874   { \__scripture_compact_end_inner: }
3875   \RenewDocumentEnvironment { narrow } { o }
3876   {
3877     \str_set:Nn \l__scripture_currenvir_str { narrow }
3878     \__scripture_reference_start_peek:
3879   }
3880   { \__scripture_compact_end_inner: }
3881   \RenewDocumentEnvironment { poetry } { o }
3882   {
3883     \str_set:Nn \l__scripture_currenvir_str { poetry }
3884     \__scripture_reference_start_peek:
3885   }
3886   { \__scripture_compact_end_inner: }
3887   \cs_set_eq:NN \ \ \unskip
3888   \cs_set_eq:NN \par \__scripture_compact_par:
3889   \cs_set_eq:NN \extraskip \relax
3890   \cs_set_eq:NN \nohang \relax
3891   \cs_set_protected:Npn \selah
3892   {
3893     \group_begin:
3894     \l__scripture_selah_font_tl
3895     \c_space_tl
3896     \__scripture_selah_format:n { \l__scripture_selah_text_tl }
3897     \c_space_tl
3898     \group_end:
3899   }
3900   \cs_set_protected_nopar:Npn \textright ##1
3901   {
3902     \unskip
3903     \c_space_tl
3904     ##1
3905   }
3906 }
3907 \__scripture_reference_start_peek:
3908 }
3909 {
3910   \bool_lazy_or:nnT
3911   { \l__scripture_compact_bool } { \l__scripture_verse_para_bool }
3912   { \cs_set_eq:NN \par \para_end: }

```

```
3913     \__scripture_end:  
3914 }
```

(End of definition for scripture and others. These functions are documented on page 12.)

```
3915 %</package>
```

Change History

v1.0			
General: First public release	1		
v1.1			
General: Add			
\scripturecurrentchapter and			
\scripturecurrentverse			
variables	7		
Add \selah function	8		
Add chapter/format option	15		
Add chapter/hide option	15		
Add chapter/showverse option	16		
Add chapter/show option	16		
Add compact option	18		
Add poetry/verse/left option	33		
Add poetry/verse/right option	33		
Add poetry/verse/sep option	33		
Add verse/hide option	26		
Add verse/sep option with small			
default	28		
Add verse/show option	28		
Add before and after hooks for			
verses and chapters	12		
Bug fixes	1		
v1.2			
General: Add \textright function	8		
Add chapter/colour option	14		
Add colour option	17		
Add language/variant option	22		
Add language option	20		
Add name/font option	22		
Add name/format option	22		
Add redletter/colour option	23		
Add redletter option	23		
Add reference/colour option	23		
Add textright/sep option	25		
Add verse/colour option	26		
Add divine name feature	6		
Add red letter feature	7		
Bug fixes	1		
v1.3			
General: Add \nofirstverse function	6		
Add \textscripture function	9		
Add added/font option	13		
Add added/format option	14		
Add inline/begin option	19		
Add inline/end option	19		
Add inline option	19		
Add reference/align option	23		
Add reference/newline option	24		
Add verse/firstformat option	26		
		Add verse/firstsep option	26
		Add verse/first option	26
		Add feature to highlight translator	
		added words	4
		Bug fixes	1
		Remove reference/delim option	13
	v2.0		
	General: Add \scripturestyle		
	function	8	
	Add inline/reference/format		
	option	20	
	Add inline/reference/sep option	20	
	Add inline/version/delim option	20	
	Add inline/version/format		
	option	20	
	Add style option	25	
	Bug fixes	1	
	Switch scripture to a L ^A T _E X list		
	based environment	1	
	v2.1		
	General: Add \heading function	5	
	Add \nopilcrow function	7	
	Add after option	14	
	Add before option	14	
	Add chapter/align option	14	
	Add chapter/para/aboveskip		
	option	16	
	Add chapter/para/belowskip		
	option	16	
	Add chapter/para/indent option	16	
	Add chapter/para option	15	
	Add chapter/smash option	16	
	Add chapter/valign option	17	
	Add chapter/xchar option	17	
	Add defaults option	18	
	Add hanging/chapter/indent		
	option	30	
	Add hanging/verse/left option	30	
	Add hanging/verse/right option	31	
	Add hanging/verse/sep option	31	
	Add heading/aboveskip option	18	
	Add heading/afterindent option	18	
	Add heading/align option	18	
	Add heading/belowskip option	18	
	Add heading/font option	19	
	Add heading/format option	19	
	Add heading/hide option	19	
	Add heading/show option	19	
	Add pilcrow/sep option	23	
	Add pilcrow option	23	

Add reference/position option . .	24	Add auto verse numbering	9
Add reference/start/inline		Bug fixes	1
option	24	Support footnotes	1
Add reference/start/newline			
option	24	v2.3	
Add reference/start/sep option .	25	General: Add mark/verse/format	
Add reference/start/vsep option	25	option	22
Add verse/para option	27	Add marks for headings and verses	13
Add version/position option . . .	29		
Add feature to add pilcrow at the		v2.4	
start of paragraphs.	23	General: Add chapter/aboveskip	
Add feature to insert simple		option	14
headings	5	Add chapter/language/variant	
Add feature to place the reference		option	15
at the start of the quotation. . . .	24	Add chapter/language option . . .	15
Add feature to set chapters in a		Add inline/font option	20
paragraph of their own	15	Add reference/language/variant	
Add feature to set each verse on a		option	24
new line.	27	Add reference/language option . .	24
Bug fixes	1	Add verse/language/variant	
		option	27
v2.2		Add verse/language option	27
General: Add book option	14	Add verse/presep option	28
Add chapter option	14	Add version/language/variant	
Add verse option	25	option	28
Add auto chapter numbering	4	Add version/language option . . .	28

Index

The italic numbers denote the pages where the corresponding entry is described, numbers underlined point to the definition, all others indicate the places where it is used.

Symbols

`\` 8, 2433, 2518, 3819, 3887
`\^` 3529, 3532

A

`\abovedisplayskip` 1921, 1922
`\abovedisplayskip` 1922
`aboveskip` (option) 13
`\added` 4, 2139
`added/font` (option) 13
`added/format` (option) 14
`\addvspace`
 .. 787, 791, 938, 1431, 1527, 1596,
 1639, 1644, 1975, 2000, 2427, 2512,
 2614, 2881, 3153, 3239, 3427, 3684
`after` (option) 14

B

`\baselineskip` 785,
 787, 857, 862, 1388, 1670, 1709,
 1921, 1948, 2427, 2512, 2666, 2881,
 3057, 3063, 3089, 3153, 3274, 3684
`before` (option) 14
`\begin` ... 985, 1080, 1109, 1174, 3731, 3734
`\belowdisplayskip` 1923
`\belowdisplayskip` 1924
`belowskip` (option) 14
`\bfseries` 14, 23, 88, 308, 557, 603
`\bigskipamount` .. 16, 18, 116, 180, 562, 573
`book` (option) 14
`bool` commands:
 `\bool_case:n` 1172, 1234, 2929
 `\bool_case:nTF` 958, 1093, 3547
 `\bool_gset:Nn` 3848
 `\bool_gset_eq:NN`
 ... 899, 902, 1597, 1602, 1647, 1652
 `\bool_gset_false:N`
 1166, 1260, 1312, 1855, 1886, 1900,
 1901, 1908, 2073, 2094, 2100, 2101,
 2102, 2320, 2553, 3625, 3712, 3834
 `\bool_gset_true:N`
 ... 736, 900, 1175, 1201, 1216,
 1228, 1270, 1295, 1428, 1598, 1649,
 1657, 1878, 2099, 2123, 2136, 2406,
 2461, 2491, 2545, 2768, 2806, 2852,
 2902, 3135, 3177, 3661, 3707, 3709
 `\bool_if:NTF` ... 675, 728, 750, 775,
 918, 953, 964, 970, 976, 987, 1008,

1025, 1042, 1053, 1064, 1067, 1069,
 1111, 1146, 1148, 1213, 1271, 1281,
 1293, 1309, 1412, 1478, 1483, 1495,
 1502, 1533, 1557, 1580, 1681, 1684,
 1792, 1798, 1804, 1854, 1872, 1902,
 1907, 1964, 1984, 1993, 2027, 2029,
 2051, 2057, 2065, 2105, 2118, 2134,
 2152, 2167, 2194, 2198, 2250, 2382,
 2430, 2467, 2515, 2554, 2556, 2559,
 2565, 2579, 2610, 2620, 2682, 2688,
 2749, 2770, 2814, 2834, 2923, 2984,
 3101, 3155, 3214, 3216, 3219, 3225,
 3235, 3423, 3435, 3475, 3495, 3604,
 3619, 3790, 3806, 3812, 3837, 3840
`\bool_if:nTF` 2124
`\bool_if_p:N` 2335
`\bool_lazy_all:nTF` 921, 1811
`\bool_lazy_and:nnTF`
 945, 2032, 2219, 2325, 2334, 2744, 3000
`\bool_lazy_any:nTF` 1618, 2201
`\bool_lazy_or:nnTF` 732, 795,
 911, 949, 991, 1115, 1198, 1209,
 1904, 2255, 2295, 2403, 2488, 2802,
 2849, 3126, 3411, 3449, 3658, 3910
`\bool_new:N` 714,
 715, 716, 717, 1224, 1290, 1771,
 1772, 1773, 1889, 2912, 3182, 3183
`\bool_not_p:n` 925, 929, 947, 1816, 2034
`\bool_set:Nn` 123
`\bool_set_eq:NN` 3551, 3559
`\bool_set_false:N`
 1778, 1783, 1794, 1795, 1968, 2106,
 3005, 3542, 3607, 3608, 3652, 3653,
 3802, 3804, 3842, 3843, 3844, 3845
`\bool_set_true:N` 954,
 1779, 2023, 2069, 2157, 2386, 2471,
 2818, 3004, 3008, 3105, 3623, 3803
`\c_false_bool` 679
`\g_tmpa_bool`
 ... 899, 902, 1597, 1602, 1648, 1653

`box` commands:

`\box_clear:N` 3035
`\box_gc_clear:N` 1251, 1267, 2103
`\box_gset_to_last:N` 1265
`\box_ht:N` 1380
`\box_ht_plus_dp:N` 3055
`\box_if_empty:NTF` 1800
`\box_move_down:nn` .. 1669, 2665, 3273

<code>\box_new:N</code>	822, 1225, 1366, 1367, 2913, 2917, 2942, 2943, 2944	
<code>\box_set_dp:Nn</code>	1680, 2676, 3284	
<code>\box_set_ht:Nn</code>	1679, 2675, 3283	
<code>\box_set_to_last:N</code>	2951, 3022	
<code>\box_set_wd:Nn</code>	1273	
<code>\box_use:N</code>	880, 1475, 1674, 2670, 2679, 3067, 3075, 3278, 3408	
<code>\box_use_drop:N</code> 1801, 3046, 3051, 3070, 3077, 3083	
<code>\box_wd:N</code>	851, 870, 1409, 2962, 3032	
<code>\break</code>	800	
C		
<code>center</code> (environment)	9, 114	
<code>center/aboveskip</code> (option)	29	
<code>center/belowskip</code> (option)	29	
<code>center/leftmargin</code> (option)	29	
<code>center/rightmargin</code> (option)	29	
<code>\centering</code>	204, 2429	
<code>\ch</code> 4, 48, 1006, 1011, 1012, 1018, 1130, 2107, 2126, 2129, 2826, 3549, 3626		
<code>\ch*</code>	4	
<code>chapter</code> (option)	14	
<code>chapter/aboveskip</code> (option)	14	
<code>chapter/align</code> (option)	14	
<code>chapter/color</code> (option)	14	
<code>chapter/colour</code> (option)	14	
<code>chapter/drop</code> (option)	14	
<code>chapter/font</code> (option)	14	
<code>chapter/format</code> (option)	15	
<code>chapter/hide</code> (option)	15	
<code>chapter/language</code> (option)	15	
<code>chapter/language/variant</code> (option)	15	
<code>chapter/nodrop</code> (option)	15	
<code>chapter/para</code> (option)	15	
<code>chapter/para/aboveskip</code> (option)	16	
<code>chapter/para/belowskip</code> (option)	16	
<code>chapter/para/indent</code> (option)	16	
<code>chapter/sep</code> (option)	16	
<code>chapter/show</code> (option)	16	
<code>chapter/showverse</code> (option)	16	
<code>chapter/smash</code> (option)	16	
<code>chapter/valign</code> (option)	17	
<code>chapter/xchar</code> (option)	17	
char commands:		
<code>\char_set_catcode_active:N</code> 3529, 3532		
<code>\clubpenalty</code> 1256, 1410, 1411, 1752, 2698		
coffin commands:		
<code>\coffin_new:N</code>	1374	
<code>\coffin_typeset:Nnnnn</code>	1539, 1548	
<code>color</code> (option)	17	
color commands:		
<code>\color_group_begin:</code>	2910	
<code>\color_select:n</code> 828, 1296, 1302, 1311, 1318, 1404, 1508, 1609, 1809, 2166, 3443	
<code>\color_set:nn</code>	2164	
<code>colour</code> (option)	17	
<code>compact</code> (option)	18	
cs commands:		
<code>\cs_generate_variant:Nn</code>	688, 1289	
<code>\cs_gset_nopar:Nn</code>	1932	
<code>\cs_gset_protected_nopar:Nn</code>	3530	
<code>\cs_if_eq:NNTF</code>	1436, 1645, 3245, 3313, 3433, 3572, 3582, 3589, 3601	
<code>\cs_if_exist_use:N</code>	170, 1284, 1286	
<code>\cs_new_protected:Nn</code> 669, 673, 695, 760, 771, 793, 823, 843, 882, 895, 941, 1164, 1191, 1226, 1249, 1279, 1291, 1307, 1323, 1376, 1382, 1426, 1490, 1574, 1664, 1697, 1738, 1758, 1774, 1840, 1846, 1868, 1890, 1898, 1919, 2025, 2062, 2241, 2316, 2380, 2443, 2465, 2528, 2551, 2570, 2737, 2812, 2886, 2945, 2991, 3018, 3099, 3160, 3188, 3203, 3212, 3230, 3418, 3464, 3499, 3515, 3536, 3617, 3691		
<code>\cs_new_protected:Npn</code>	2906	
<code>\cs_new_protected_nopar:Nn</code>	721, 1953, 1960, 2014, 2021, 2794, 2920	
<code>\cs_set_eq:NN</code> 710, 753, 754, 756, 1011, 1018, 1028, 1035, 1045, 1047, 1056, 1058, 1986, 1989, 2064, 2095, 2096, 2097, 2115, 2139, 2140, 2141, 2144, 2151, 2158, 2236, 2249, 2313, 2328, 2373, 2399, 2432, 2445, 2484, 2517, 2530, 2835, 2836, 2837, 2844, 2919, 3118, 3156, 3162, 3497, 3513, 3533, 3540, 3544, 3597, 3603, 3634, 3635, 3646, 3647, 3796, 3798, 3814, 3815, 3816, 3817, 3818, 3828, 3829, 3839, 3846, 3871, 3887, 3888, 3889, 3890, 3912	
<code>\cs_set_nopar:Npn</code>	711, 712	
<code>\cs_set_protected:Npn</code> 1253, 2098, 2433, 2518, 3819, 3891	
<code>\cs_set_protected_nopar:Npn</code> 725, 2142, 2143, 2145, 3636, 3900	
<code>\currentgrouplevel</code>	1417, 1421, 1485, 1691, 1702, 1743, 1763, 2375, 2400, 2485, 2846, 3096, 3119, 3233, 3655	
D		
<code>\DeclareDocumentCommand</code> 2107, 2116, 2132, 2826, 3626	
<code>\DeclareDocumentEnvironment</code>	3742, 3748, 3754, 3760, 3766, 3772, 3784	

<code>\group_end:</code> .	841, 1477, 1570, 1613, 1640, 1660, 1824, 1829, 1838, 1958, 1983, 2019, 2681, 2982, 3210, 3410, 3446, 3461, 3494, 3512, 3535, 3898
H	
<code>hanging</code> (environment)	10, 114
<code>hanging/aboveskip</code> (option)	30
<code>hanging/belowskip</code> (option)	30
<code>hanging/chapter/indent</code> (option)	30
<code>hanging/hang</code> (option)	30
<code>hanging/leftmargin</code> (option)	30
<code>hanging/rightmargin</code> (option)	30
<code>hanging/verse/left</code> (option)	30
<code>hanging/verse/right</code> (option)	31
<code>hanging/verse/sep</code> (option)	31
<code>hbox</code> commands:	
<code>\hbox:n</code>	812, 2302, 3195
<code>\hbox_overlap_center:n</code>	1604
<code>\hbox_overlap_left:n</code>	1672, 2668, 2751, 2757, 2772, 2778, 3276, 3437, 3477, 3483
<code>\hbox_overlap_right:n</code>	2759, 2780, 3485
<code>\hbox_set:Nn</code>	845, 1378, 1396, 1667, 2663, 2952, 3023, 3271
<code>\hbox_to_wd:nn</code>	2360, 2369, 2799
<code>\hbox_unpack:N</code>	3025
<code>\hbox_unpack_drop:N</code>	2954, 2970, 2979
<code>hcoffin</code> commands:	
<code>\hcoffin_set:Nn</code>	1505
<code>\heading</code>	5, 13, 19, 21, 960, 1095, 2140
<code>heading/aboveskip</code> (option)	18
<code>heading/afterindent</code> (option)	18
<code>heading/align</code> (option)	18
<code>heading/belowskip</code> (option)	18
<code>heading/font</code> (option)	19
<code>heading/format</code> (option)	19
<code>heading/hide</code> (option)	19
<code>heading/show</code> (option)	19
<code>\hfil</code>	809, 3192
<code>\hfill</code>	802, 814, 866, 915, 3197
<code>hook</code> commands:	
<code>\hook_gclear_next_code:n</code>	1488, 1572, 2147, 2148, 2149, 2292, 2293, 2318, 2319, 2845, 3468, 3648, 3708
<code>\hook_gput_code:nnn</code>	708, 2150
<code>\hook_gput_next_code:nn</code>	748, 989, 1113, 1266, 1299, 1315, 1699, 1719, 1740, 1749, 1760, 1765, 1879, 1894, 1995, 2036, 2043, 2196, 2341, 2462, 2546, 2690, 2696, 2702, 2730, 2903, 3178, 3308, 3541, 3649, 3710
<code>\hook_gremove_code:nn</code>	2312
<code>\hook_if_empty:nTF</code>	1850, 1892
<code>\hook_new:n</code>	691, 692, 3181
<code>\hook_new_pair:nn</code>	689, 690
<code>\hook_use:n</code>	1474, 1476, 1532, 1556, 1566, 1567, 1606, 1614, 1634, 1635, 1641, 1661, 1803, 1836, 1837, 1926, 2678, 2680, 3407, 3409, 3416, 3439, 3448, 3457, 3458, 3462, 3496, 3526
<code>hooks</code> :	
<code>scripture/chap/after</code>	12, 47
<code>scripture/chap/before</code>	12, 47
<code>scripture/pilcrow</code>	47
<code>scripture/poetry/para/after</code>	102
<code>scripture/poetry/pilcrow</code>	47
<code>scripture/verse/after</code>	12, 47
<code>scripture/verse/before</code>	12, 47
<code>\hspace</code>	3015
I	
<code>\iflanguageloaded</code>	1338
<code>\ignorespaces</code>	2004, 3809
<code>indent</code> (option)	19
<code>inline</code> (option)	19
<code>inline/begin</code> (option)	19
<code>inline/end</code> (option)	19
<code>inline/font</code> (option)	20
<code>inline/reference/format</code> (option)	20
<code>inline/reference/sep</code> (option)	20
<code>inline/version/delim</code> (option)	20
<code>inline/version/format</code> (option)	20
<code>int</code> commands:	
<code>\int_compare:nNnTF</code>	779, 860, 1701, 1706, 1742, 1746, 1762, 2288, 2426, 2511, 2693, 2704, 2880, 3037, 3152, 3310, 3335, 3470, 3503, 3683
<code>\int_compare_p:nNn</code>	2337
<code>\int_gincr:N</code>	2120, 2135
<code>\int_gset:Nn</code>	1420, 1936, 2074, 2075, 2121, 2374, 3095
<code>\int_gset_eq:NN</code>	1416, 1485, 1691, 1745, 2045, 2330, 2400, 2485, 2692, 2846, 3092, 3119, 3233, 3655
<code>\int_gzero:N</code>	789, 1731, 1766, 2038, 2039, 2088, 2428, 2513, 2734, 2882, 3154, 3685
<code>\int_new:N</code>	718, 1371, 1694, 1695, 1696, 2916, 2918, 3187
<code>\int_set:Nn</code>	1935, 2090, 2092, 2276, 3059, 3518
<code>\int_set_eq:NN</code>	1410, 1751, 2091, 2093, 2698, 2948, 2971
<code>\int_use:N</code>	1935, 1937, 2083, 2085, 2126, 2129, 2137, 3518
<code>\int_zero:N</code>	817, 3088, 3200

<code>\c_one_int</code>	61, 371, 554, 618, 780, 861, 1707, 1747, 2338, 2375, 2426, 2511, 2694, 2705, 2880, 3096, 3152, 3311, 3336, 3470, 3503, 3683
<code>\c_zero_int</code>	2288
<code>\item</code>	2237, 2440, 2525, 2883, 3157, 3688
<code>\itemindent</code>	49, 2171, 2177, 2396, 2410, 2481, 2495, 2841, 2860, 3115, 3130, 3136, 3643, 3665
<code>\itshape</code>	19, 25, 215, 354, 577, 614
K	
<code>\kern</code>	1511, 1825, 1830, 1911, 2372
keys commands:	
<code>\l_keys_choice_str</code>	70, 146, 189, 298, 326, 436
<code>\keys_define:nn</code>	29, 684
<code>\keys_precompile:nnN</code>	543
<code>\keys_set:nn</code>	2079, 2390, 2475, 2822, 2996, 3109, 3721, 3776
L	
<code>\labelsep</code>	2407, 2492, 2853, 3122, 3662
<code>language</code> (option)	20
<code>language/variant</code> (option)	22
<code>\LastMark</code>	13
<code>leftmargin</code> (option)	22
<code>\leftmargin</code>	904, 1590, 1593, 2175, 2408, 2493, 2577, 2582, 2589, 2594, 2599, 2604, 2642, 2718, 2724, 2854, 3123, 3663
<code>\leftprotrusion</code>	60, 1284, 1286
<code>\leftskip</code>	2967, 2978, 3010
legacy commands:	
<code>\legacy_if:nTF</code>	723, 1193, 1438, 1497, 1704, 1777, 1857, 1874, 1969, 2170, 2217, 2394, 2479, 2796, 2839, 3113, 3641
<code>\legacy_if_p:n</code>	2745
<code>\legacy_if_set:nn</code>	1625, 1991
<code>\legacy_if_set_false:n</code>	1259, 1269, 2309, 2459, 2543, 2900, 3175, 3705
<code>\legacy_if_set_true:n</code>	1252, 1629
<code>\linewidth</code>	24, 319, 855, 874, 1455, 1467, 1472, 1571, 1582, 1733, 1862, 1880, 1944, 1966, 2308, 2345, 2354, 2358, 2447, 2463, 2531, 2547, 2594, 2604, 2607, 2724, 2727, 2729, 2731, 2733, 2904, 2967, 3015, 3179, 3298, 3302, 3346, 3356, 3393, 3401, 3521, 3525, 3713
<code>\list</code>	2173, 2401, 2486, 2847, 3120, 3656
<code>\listparindent</code>	2181, 2411, 2496, 2865, 3125, 3508, 3510, 3669
<code>\LORD</code>	6, 48, 711, 2142
M	
mark commands:	
<code>\mark_get_marks_for_reinsertion:nnN</code>	3066, 3074
<code>\mark_insert:nn</code>	699, 1981, 2003
<code>\mark_new_class:n</code>	693, 694
<code>mark/verse/format</code> (option)	22
marks:	
<code>scripture/heading</code>	13, 47
<code>scripture/verse</code>	13, 47
<code>\medskipamount</code>	16, 18, 29, 30, 32, 33, 120, 174, 212, 442, 445, 454, 457, 467, 470, 503, 506, 515, 518, 563, 571, 576, 638, 639, 642, 643, 646, 647, 655, 656, 659, 660
<code>midparachap</code> (environment)	10, 115
<code>midparachap/hideverse</code> (option)	31
<code>midparachap/showverse</code> (option)	32
mode commands:	
<code>\mode_if_horizontal:TF</code>	2435, 2520, 3821
<code>\mode_if_inner:TF</code>	3048
<code>\mode_if_vertical:TF</code>	773, 1136, 1141, 1499, 1595, 1780, 1842, 1848, 1859, 1870, 1876, 2739, 3538
<code>\mode_if_vertical_p:</code>	2404, 2489, 2850, 3127, 3659, 3849
<code>\mode_leave_vertical:</code>	1535, 1781, 2630, 3253
msg commands:	
<code>\msg_error:nn</code>	2067, 2384, 2469, 2816, 3103, 3621
<code>\msg_error:nnn</code>	362, 1351, 2008
<code>\msg_new:nnnn</code>	6, 10, 14, 18, 24
<code>\msg_warning:nnn</code>	1358
N	
<code>\name</code>	6, 48, 710, 2141, 2142, 2143
<code>name/font</code> (option)	22
<code>name/format</code> (option)	22
<code>narrow</code> (environment)	11, 115
<code>narrow/aboveskip</code> (option)	32
<code>narrow/belowskip</code> (option)	32
<code>narrow/leftmargin</code> (option)	32
<code>narrow/rightmargin</code> (option)	32
<code>\NeedsTeXFormat</code>	3
<code>\newch</code>	4, 14, 1023, 1028, 1029, 1035, 1132, 2116, 3557
<code>\newch*</code>	4
<code>\NewDocumentCommand</code>	3719, 3723, 3727
<code>\NewDocumentEnvironment</code>	3739

\newvs	9, 25, 734, 1051, 1056, 1057, 1058, 1139, 1184, 1200, 1211, 1242, 2132, 2804, 2935, 3586	colour	17
\nobreak	813, 936, 1554, 1617, 1832, 1912, 1998, 3196	compact	18
\nofirstverse	6, 2144	defaults	18
\nohang	6, 968, 1099, 2837, 3890	extraskip	18
noindent (option)	22	flushright/aboveskip	29
\noindent	49, 58, 725, 753, 754, 756, 901, 962, 1097, 1436, 1445, 1601, 1650, 1986, 1989, 2064, 2151, 2228, 2249, 2294, 2313, 2977, 3040, 3050, 3069, 3082, 3090, 3248, 3434, 3573, 3583, 3590, 3602, 3647, 3818, 3828, 3839	flushright/belowskip	29
\nointerlineskip	2959, 3036	flushright/leftmargin	29
\nopilcrow	7, 974, 1101, 2098, 3569	flushright/rightmargin	29
O		font	18
\obeylines	3	hanging/aboveskip	30
options:		hanging/belowskip	30
aboveskip	13	hanging/chapter/indent	30
added/font	13	hanging/hang	30
added/format	14	hanging/leftmargin	30
after	14	hanging/rightmargin	30
before	14	hanging/verse/left	30
belowskip	14	hanging/verse/right	31
book	14	hanging/verse/sep	31
center/aboveskip	29	heading/aboveskip	18
center/belowskip	29	heading/afterindent	18
center/leftmargin	29	heading/align	18
center/rightmargin	29	heading/belowskip	18
chapter	14	heading/font	19
chapter/aboveskip	14	heading/format	19
chapter/align	14	heading/hide	19
chapter/color	14	heading/show	19
chapter/colour	14	indent	19
chapter/drop	14	inline	19
chapter/font	14	inline/begin	19
chapter/format	15	inline/end	19
chapter/hide	15	inline/font	20
chapter/language	15	inline/reference/format	20
chapter/language/variant	15	inline/reference/sep	20
chapter/nodrop	15	inline/version/delim	20
chapter/para	15	inline/version/format	20
chapter/para/aboveskip	16	language	20
chapter/para/belowskip	16	language/variant	22
chapter/para/indent	16	leftmargin	22
chapter/sep	16	mark/verse/format	22
chapter/show	16	midparachap/hideverse	31
chapter/showverse	16	midparachap/showverse	32
chapter/smash	16	name/font	22
chapter/valign	17	name/format	22
chapter/xchar	17	narrow/aboveskip	32
color	17	narrow/belowskip	32
		narrow/leftmargin	32
		narrow/rightmargin	32
		noindent	22
		parindent	22
		parskip	22
		pilcrow	23
		pilcrow/sep	23
		poetry/aboveskip	33
		poetry/belowskip	33

poetry/bigindent	33	P	
poetry/indent	33	\P	1910
poetry/leftmargin	33	\pagegoal	3057, 3063
poetry/rightmargin	33	\pagetotal	3057, 3063
poetry/verse/left	33	\paperwidth	1941
poetry/verse/right	33	\par	9,
poetry/verse/sep	33		18, 19, 58, 59, 777, 934, 1253, 1258,
redletter	23		1686, 1881, 2328, 2373, 2399, 2432,
redletter/color	23		2445, 2484, 2517, 2530, 2609, 2844,
redletter/colour	23		2950, 2981, 3007, 3020, 3086, 3118,
reference/align	23		3156, 3162, 3646, 3815, 3888, 3912
reference/color	23	para commands:	
reference/colour	23	\para_end:	776, 804, 818, 919, 933,
reference/font	23		1214, 1499, 1595, 1616, 1685, 1861,
reference/format	24		2328, 2399, 2445, 2484, 2530, 2844,
reference/language	24		3118, 3162, 3517, 3646, 3833, 3912
reference/language/variant	24	\para_omit_indent:	
reference/newline	24		1002, 1126, 1721, 1996, 2199, 2230
reference/position	24	\parfillskip	816, 2980, 3199
reference/sep	24	parindent (option)	22
reference/start/inline	24	\parindent	22, 88, 93,
reference/start/newline	24		276, 595, 1167, 1229, 1717, 1852,
reference/start/sep	25	\parsep	1971, 2838, 3125, 3131, 3132, 3687
reference/start/vsep	25		2183, 2412, 2497, 2866, 3139, 3670
rightmargin	25	\parshape	3, 70, 102,
selah/font	25		105, 106, 1446, 1571, 1733, 1862,
selah/format	25		1880, 1966, 2308, 2344, 2463, 2547,
selah/sep	25		2586, 2715, 2729, 2731, 2733, 2904,
selah/text	25		3179, 3292, 3338, 3385, 3519, 3713
style	25	parskip (option)	22
textright/sep	25	\parskip	16, 22, 279,
verse	25		596, 783, 1579, 1709, 1948, 2184,
verse/color	26		2190, 2275, 2412, 2415, 2421, 2497,
verse/colour	26		2500, 2506, 2866, 2869, 2875, 3016,
verse/first	26		3089, 3139, 3142, 3148, 3672, 3678
verse/firstformat	26	\partopsep	2182, 2413, 2498, 2867, 3140, 3671
verse/firstsep	26	\peek	60
verse/font	26	peek commands:	
verse/format	26	\peek_analysis_map_break:n	744,
verse/hide	26		979, 1086, 1103, 1156, 1187, 1203,
verse/language	27		1220, 1245, 2809, 2938, 3555, 3563,
verse/language/variant	27		3566, 3568, 3574, 3584, 3591, 3610
verse/para	27	\peek_analysis_map_inline:n	
verse/presep	28		730, 956, 1091, 1170,
verse/sep	28		1196, 1207, 1232, 2800, 2927, 3545
verse/show	28	\peek_meaning:NTF	1637
version	28	\peek_remove_spaces:n	
version/delim	28		943, 1168, 1230, 2925
version/format	28	\penalty	810, 1748, 2695, 3193
version/language	28	pilcrow (option)	23
version/language/variant	28	pilcrow/sep (option)	23
version/position	29	poetry (environment)	11, 115
		poetry/aboveskip (option)	33
		poetry/belowskip (option)	33

poetry/bigindent (option) 33
 poetry/indent (option) 33
 poetry/leftmargin (option) 33
 poetry/rightmargin (option) 33
 poetry/verse/left (option) 33
 poetry/verse/right (option) 33
 poetry/verse/sep (option) 33
 \poetryaux 3787
 \postdisplaypenalty 2091, 2276
 \predisplaypenalty 2090
 \predisplaysize 1927, 1930, 1934
 \prevdepth 96, 3021, 3087
 \prevgraf 71, 96, 1745, 1935,
 2046, 2288, 2692, 3038, 3088, 3518
 \ProcessKeyOptions 541
 \ProvidesExplPackage 4

R

\raggedleft 199, 2514
 \raggedright 194, 3686
 redletter (option) 23
 redletter/color (option) 23
 redletter/colour (option) 23
 \redletteroff 7, 2097
 \redletteron 7, 2096
 reference/align (option) 23
 reference/color (option) 23
 reference/colour (option) 23
 reference/font (option) 23
 reference/format (option) 24
 reference/language (option) 24
 reference/language/variant (option) 24
 reference/newline (option) 24
 reference/position (option) 24
 reference/sep (option) 24
 reference/start/inline (option) 24
 reference/start/newline (option) 24
 reference/start/sep (option) 25
 reference/start/vsep (option) 25
 \relax 9, 18, 19, 2238, 2373, 2441,
 2526, 2884, 3158, 3689, 3889, 3890
 \renewcommand* 7
 \RenewDocumentCommand
 1012, 1029, 1046, 1057
 \RenewDocumentEnvironment 3808, 3825,
 3850, 3856, 3862, 3868, 3875, 3881
 rightmargin (option) 25
 \rightmargin 1590,
 2176, 2409, 2494, 2859, 3124, 3664
 \rightskip 2967, 3011

S

scripture (environment) 12, 114

scripture commands:

\added 4, 13, 14
 \ch 7, 14, 16, 17, 24, 31, 32, 50, 102, 113
 \extraskip 18
 \GOD 6
 \heading 3, 5, 18, 19, 34, 47, 119
 \LORD 6
 \name 6, 22
 \nofirstverse 26, 119
 \nopilcrow 23, 119
 \redletteroff 7, 23
 \redletteron 7, 23
 \scripturecurrentchapter 7, 47, 119
 \scripturecurrentverse 7, 47, 119
 \scripturesetup 13, 29–32
 \scripturestyle 25, 119
 \selah 25, 119
 \textright 25, 119
 \textscripture 9, 19, 20, 119
 \vs 6,
 7, 16, 24, 26, 28, 31–33, 47, 59, 92, 113

scripture internal commands:

\l__scripture_aboveskip_tl
 31, 2184, 2190, 3792
 \l__scripture_active_bool
 714, 2065, 2069
 \l__scripture_active_inner_bool
 714, 2382, 2386, 2467, 2471,
 2814, 2818, 3101, 3105, 3619, 3623
 __scripture_added:n 1953, 1953, 2139
 \l__scripture_added_font_tl 34, 1956
 __scripture_added_format:n 37, 1957
 \l__scripture_after_tl 40, 2314
 \l__scripture_before_tl 43, 2086
 __scripture_begin:nn 2062, 2062, 3741
 \l__scripture_belowskip_tl
 46, 2279, 2285, 3793
 \g__scripture_book_tl 49, 2076, 2081
 __scripture_calc_final_line_
 length: 1919,
 1919, 2277, 2329, 2888, 3163, 3693
 \l__scripture_center_aboveskip_
 tl 440, 2415, 2421
 __scripture_center_begin:n
 2380, 2380, 3744
 \l__scripture_center_belowskip_
 tl 443, 2449, 2455
 __scripture_center_end:
 2443, 2443, 3747
 \l__scripture_center_leftmargin_
 tl 446, 2408
 \l__scripture_center_rightmargin_
 tl 449, 2409

__scripture_ch_output:nn	\l__scripture_chap_tmp_box
. 2025, 2025, 2113 1367, 1378,
__scripture_chap:n . 1664, 1664, 2031	1380, 1396, 1409, 1674, 2670, 3278
\l__scripture_chap_box 1366,	l__scripture_chap_tmp_box 1366
1475, 1667, 1679, 1680, 2663, 2675,	\g__scripture_chap_width_dim
2676, 2679, 3271, 3283, 3284, 3408 1370, 1408, 1450, 1456, 1462,
\l__scripture_chap_drop_bool	1468, 1713, 1724, 2349, 2355, 2363,
. 93, 108, 2029,	2575, 2582, 2589, 2594, 2599, 2604,
2106, 2326, 2556, 3002, 3216, 3842	2632, 2638, 2650, 2659, 2711, 2719,
\l__scripture_chap_drop_llap_-	2724, 2963, 3042, 3044, 3257, 3268,
bool 3843	3287, 3304, 3317, 3330, 3364, 3379
\l__scripture_chap_drop_smash_-	\l__scripture_chap_X_height_dim
bool 3844	. . 1372, 1379, 1388, 1394, 1518, 1523
\l__scripture_chap_font_tl	\l__scripture_chapter_aboveskip_-
. 86, 1398, 1509, 1610, 3441	tl . . 62, 1429, 1433, 1525, 1529,
\l__scripture_chap_fontsize_dim	2612, 2616, 3237, 3241, 3425, 3429
. 1368, 1389, 1400, 1401	\l__scripture_chapter_align_tl .
__scripture_chap_format:n 68, 69, 74, 79, 1540, 1549
. 96, 1406, 1510, 1612, 3445	\l__scripture_chapter_colour_tl
\l__scripture_chap_height_dim 89, 1403, 1404,
. 1369, 1387, 1393	1507, 1508, 1608, 1609, 3442, 3443
\g__scripture_chap_par_prevgraf_-	\g__scripture_chapter_int
int 71, 780, 789, 59, 2074, 2083, 2120, 2126, 2129
861, 1371, 1707, 1731, 1745, 1747,	\l__scripture_chapter_language_-
1766, 1936, 1937, 2038, 2088, 2332,	tl 101
2426, 2428, 2511, 2513, 2692, 2694,	\l__scripture_chapter_language_-
2705, 2734, 2880, 2882, 3093, 3152,	variant_tl 105
3154, 3336, 3470, 3503, 3683, 3685	\l__scripture_chapter_valign_tl
__scripture_chap_par_save_- 144, 145,
prevgraf_at_group_level:	150, 155, 160, 1514, 1520, 1541, 1550
. 1423, 1486, 1738, 1738, 1755	\l__scripture_clubpenalty_saved_-
\l__scripture_chap_para_aboveskip_-	int 718, 1410, 1752, 2698
tl 114, 1596	\l__scripture_colour_tl 52, 2164, 2165
\l__scripture_chap_para_afterindent_-	\l__scripture_compact_bool
bool 124, 1626 56, 918,
\l__scripture_chap_para_belowskip_-	953, 1067, 1146, 1281, 1905, 2034,
tl 118, 1639, 1644	2198, 2335, 2923, 3803, 3837, 3911
\l__scripture_chap_para_bool	__scripture_compact_end_inner:
. 110, 2051, 2057, 2105, 2559, 2316, 2316,
2565, 3219, 3225, 3802, 3806, 3845	3855, 3861, 3867, 3874, 3880, 3886
\l__scripture_chap_sep_tl	__scripture_compact_par:
. 129, 1451, 1457, 1463, 1469, 1164, 1164, 3888
1511, 1675, 1714, 1725, 2350, 2356,	\g__scripture_current_group_-
2364, 2572, 2964, 3042, 3044, 3255,	level_a_int
3257, 3318, 3331, 3364, 3380, 3447 1691, 1694, 1702, 2374, 3095
\l__scripture_chap_show_bool	\g__scripture_current_group_-
. 99, 132, 1502, 1580, 2027,	level_b_int
2554, 2610, 3214, 3235, 3423, 3435 1417, 1420, 1485, 1695,
\l__scripture_chap_show_verse_-	1743, 2400, 2485, 2846, 3119, 3655
bool 135, 1478,	\g__scripture_current_group_-
1557, 2682, 3001, 3005, 3412, 3450	level_c_int 1694, 1763, 3233
\l__scripture_chap_smash_bool	\l__scripture_currenvir_str
. 138, 1533 741, 752, 796, 797, 903, 912,
	913, 926, 930, 950, 951, 994, 999,

1074, 1118, 1123, 1152, 1176, 1283,
 1297, 1300, 1313, 1316, 1414, 1562,
 1583, 1599, 1620, 1621, 1622, 1656,
 1688, 1818, 1951, 1962, 2011, 2072,
 2205, 2209, 2213, 2222, 2342, 2387,
 2472, 2819, 2993, 3106, 3624, 3827,
 3852, 3858, 3864, 3870, 3877, 3883
 \l__scripture_default_options_tl
 542, 668, 671
 __scripture_doendpe: 1249, 1249, 3817
 __scripture_dopeend: 59
 __scripture_drop_chap_output: .
 1426, 1426, 1687, 2969, 2983
 __scripture_drop_chap_set_up:n
 1382, 1382, 1666, 2573, 3232
 __scripture_end: .. 2241, 2241, 3913
 __scripture_extra_skip:
 771, 771, 2095
 \l__scripture_extraskip_tl
 172, 783, 791
 \g__scripture_final_line_dim ...
 852, 871, 877, 1918,
 1929, 1941, 1943, 1946, 2278, 2362,
 2369, 2446, 2531, 3030, 3043, 3091
 \g__scripture_first_verse_set_-
 bool 716, 2073, 2118, 2123, 2134, 2136
 __scripture_fix_protrusion:n ..
 982, 1087, 1106,
 1158, 1279, 1279, 1289, 3576, 3612
 \l__scripture_flushright_-
 aboveskip_tl 452, 2500, 2506
 __scripture_flushright_begin:n
 2465, 2465, 3750
 \l__scripture_flushright_-
 belowskip_tl 455, 2533, 2539
 __scripture_flushright_end: ...
 2528, 2528, 3753
 \l__scripture_flushright_-
 leftmargin_tl 458, 2493
 \l__scripture_flushright_-
 rightmargin_tl 462, 2494
 \l__scripture_font_tl 175, 2154
 __scripture_format_full_ref:nn
 823,
 823, 847, 888, 891, 907, 910, 2264, 2269
 __scripture_get_X_height:
 1376, 1376, 1386, 1504
 \l__scripture_hanging_aboveskip_-
 tl 465, 2869, 2875
 __scripture_hanging_begin:n ...
 2812, 2812, 3756
 \l__scripture_hanging_belowskip_-
 tl 468, 2890, 2896
 __scripture_hanging_ch:n
 2551, 2551, 2832
 __scripture_hanging_chap:n
 2557, 2570, 2570
 \l__scripture_hanging_chap_sep_-
 dim 2550,
 2572, 2575, 2581, 2590, 2595, 2600,
 2605, 2622, 2624, 2632, 2639,
 2649, 2656, 2671, 2712, 2720, 2725
 \l__scripture_hanging_chapter_-
 indent_bool 472, 2579, 2620
 __scripture_hanging_end:
 2886, 2886, 3759
 \l__scripture_hanging_hang_tl ..
 475, 2644, 2797, 2857, 2862, 2865, 2878
 \l__scripture_hanging_leftmargin_-
 tl 478, 2634,
 2651, 2658, 2708, 2763, 2784, 2856
 \l__scripture_hanging_parindent_-
 saved_dim .. 2549, 2625, 2799, 2838
 \l__scripture_hanging_rightmargin_-
 tl 481, 2859
 __scripture_hanging_verse:n ...
 1887, 2737, 2737, 2836
 __scripture_hanging_verse_-
 para:n 1868, 1868, 2835
 \l__scripture_hanging_verse_-
 right_bool ... 485, 488, 2749, 2770
 \l__scripture_hanging_verse_sep_-
 tl 491, 2743
 __scripture_heading:n
 1960, 1960, 2140
 \l__scripture_heading_aboveskip_-
 tl 178, 1975
 \l__scripture_heading_afterindent_-
 bool 181, 1992, 1993
 \l__scripture_heading_align_tl .
 187, 188, 193, 198, 203, 1978
 \l__scripture_heading_belowskip_-
 tl 210, 2000
 \l__scripture_heading_font_tl ..
 213, 1979
 __scripture_heading_format:n ..
 216, 1980
 \l__scripture_heading_show_bool
 219, 221, 1964, 3804
 \l__scripture_indent_bool
 224, 272, 987, 1111, 2194
 \l__scripture_inline_begin_tl ..
 230, 2168
 \l__scripture_inline_bool
 227, 2152, 2167, 2250, 3790
 \l__scripture_inline_end_tl 233, 2254

\l__scripture_inline_font_tl ...	\l__scripture_mid_para_chap_-
..... 236, 2153	split_box 2942, 3061, 3065
__scripture_inline_ref_format:n	\l__scripture_mid_para_chap_-
..... 240, 3796	split_top_box 2942, 3064, 3067, 3070
\l__scripture_inline_ref_sep_tl	\l__scripture_mid_para_chap_-
..... 243, 3797	totalleftmargin_dim
\l__scripture_inline_version_- 2915, 3013, 3033
delim_tl 246, 3801	__scripture_mid_para_drop_chap_-
__scripture_inline_version_-	output: 1682, 2945, 2945
format:n 250, 3799	\l__scripture_midparachap_show_-
__scripture_insert_verse_mark:	verse_bool
..... 695, 695, 1482, 1561, 1835, 495, 498, 2984, 3004, 3553, 3561
2686, 2789, 2988, 3415, 3453, 3492	\g__scripture_mode_vertical_-
\l__scripture_language_tl	inner_bool ... 717, 1069, 1148,
..... 1325, 1329,	1166, 1175, 1901, 2102, 2320, 3848
1333, 1338, 1342, 1347, 1354, 1361	__scripture_name:n
\l__scripture_language_variant_- 710, 711, 712, 2014, 2014, 2141
tl 1327, 1340, 1346	\l__scripture_name_font_tl 266, 2017
\l__scripture_leftmargin_tl	__scripture_name_format:n 269, 2018
..... 259, 2175, 3794	\l__scripture_narrow_aboveskip_-
__scripture_list_para_end:	tl 501, 3142, 3148
..... 1974, 1982, 2236	__scripture_narrow_begin:n
\l__scripture_main_language_tl . 253 3099, 3099, 3768
\l__scripture_main_language_-	\l__scripture_narrow_belowskip_-
variant_tl 256	tl 504, 3165, 3171
__scripture_mark_verse_format:nnn	__scripture_narrow_end:
..... 262, 701 3160, 3160, 3771
__scripture_mid_para_chap_-	\l__scripture_narrow_leftmargin_-
begin:n 2991, 2991, 3762	tl 507, 3123
\l__scripture_mid_para_chap_bool	\l__scripture_narrow_rightmargin_-
..... 1681, 2912, 3008	tl 510, 3124
\l__scripture_mid_para_chap_box	__scripture_no_first_verse: ...
..... 2942, 3009, 2021, 2021, 2144
3051, 3055, 3062, 3075, 3077, 3083	\l__scripture_no_verse_first_-
__scripture_mid_para_chap_end:	bool 1772, 1792, 1795, 2023
..... 3018, 3018, 3765	__scripture_nodrop_chap:n
__scripture_mid_para_chap_end:n	.. 1490, 1490, 2053, 2059, 2561, 2567
..... 2919, 2920, 3829, 3871	\l__scripture_nodrop_chap_coffin
__scripture_mid_para_chap_end_- 1374, 1505, 1539, 1548
orig:n 2919, 2919, 2922	\l__scripture_nodrop_chap_-
\l__scripture_mid_para_chap_-	voffset_dim
line_box 1375, 1513, 1517, 1522, 1543, 1552
.. 2913, 2951, 2954, 3022, 3025, 3046	__scripture_nohang:
\g__scripture_mid_para_chap_- 2627, 2794, 2794, 2837
prevdepth_dim 2914, 3021, 3087	__scripture_noindent_saved: ...
\g__scripture_mid_para_chap_- 753, 756, 1195, 1215, 1442,
prevgraf_int	1600, 1882, 1976, 1989, 2064, 2151,
.. 2039, 2046, 2331, 2338, 2916, 3094	2249, 2313, 2333, 2798, 3647, 3828
\l__scripture_mid_para_chap_-	__scripture_obeylines:
snap_box 2917, 2952, 115, 3528, 3530, 3786
2962, 2970, 2979, 3023, 3032, 3035	__scripture_obeylines_eol:
\l__scripture_mid_para_chap_- 3533, 3536, 3595
spacefactor_int .. 2918, 2948, 2971	\l__scripture_outer_itemindent_-
	dim 720, 2171,

2172, 2179, 2396, 2398, 2410, 2481,
 2483, 2495, 2841, 2843, 2863, 3115,
 3117, 3131, 3136, 3643, 3645, 3667
 _scripture_para_chap:n
 1574, 1574,
 2052, 2058, 2560, 2566, 3220, 3226
 \l_scripture_para_chap_indent_-
 dim 1373, 1582, 1585,
 1586, 1589, 1591, 1592, 1593, 1603
 \l_scripture_para_indent_tl ...
 403, 1275, 1852
 \g_scripture_para_mode_vertical_-
 bool 1224, 1228,
 1900, 2405, 2490, 2851, 3128, 3660
 _scripture_para_noindent:
 754, 1191, 1191, 1436, 1986, 3818, 3839
 _scripture_para_par:
 59, 1226, 1226,
 1261, 2432, 2517, 3156, 3815, 3816
 \l_scripture_parindent_tl
 274, 405, 628, 1167, 1229,
 1440, 2161, 2162, 2179, 2181, 2193
 \l_scripture_parskip_tl .. 277, 2183
 \l_scripture_pilcrow_bool
 280, 1902, 1968, 3608, 3652
 _scripture_pilcrow_output: ...
 742, 1070, 1075, 1149,
 1153, 1179, 1181, 1202, 1219, 1237,
 1239, 1443, 1563, 1689, 1895, 1898,
 1898, 2687, 2787, 2932, 3606, 3651
 \l_scripture_pilcrow_sep_tl ...
 283, 1911
 \l_scripture_poetry_aboveskip_-
 tl 513, 3672, 3678
 _scripture_poetry_begin:
 3617, 3617, 3780
 \l_scripture_poetry_belowskip_-
 tl 516, 3695, 3701
 \l_scripture_poetry_big_indent_-
 tl 519, 3300, 3302,
 3352, 3357, 3398, 3401, 3523, 3525
 _scripture_poetry_ch:n
 3212, 3212, 3632
 _scripture_poetry_chap:n
 3217, 3230, 3230
 \l_scripture_poetry_chap_-
 indent_dim
 .. 3184, 3247, 3251, 3261, 3267, 3290
 \l_scripture_poetry_chap_-
 parshape_correction_dim 3186,
 3285, 3295, 3298, 3324, 3328,
 3342, 3347, 3373, 3377, 3389, 3394
 \l_scripture_poetry_chap_sep_-
 dim 3184, 3255, 3264, 3279, 3288, 3304
 _scripture_poetry_end:
 3691, 3691, 3783
 \g_scripture_poetry_end_bool ..
 2094, 3182, 3709, 3712
 \l_scripture_poetry_indent_tl ..
 522, 3252,
 3369, 3382, 3667, 3669, 3681, 3687
 \l_scripture_poetry_leftmargin_-
 tl .. 525, 3260, 3266, 3289, 3306,
 3321, 3332, 3368, 3381, 3489, 3663
 \g_scripture_poetry_midparachap_-
 show_verse_bool 3625
 \l_scripture_poetry_midparachap_-
 show_verse_bool
 .. 3183, 3413, 3451, 3542, 3552, 3560
 _scripture_poetry_mode_-
 horizontal_verse:n 3499, 3499, 3544
 _scripture_poetry_mode_-
 vertical_verse:n
 1645, 3245, 3313,
 3433, 3464, 3464, 3505, 3509, 3540,
 3572, 3582, 3589, 3597, 3601, 3634
 _scripture_poetry_nodrop_-
 chap:n 3221, 3227, 3418, 3418
 _scripture_poetry_par:
 3515, 3515,
 3554, 3562, 3571, 3581, 3588, 3600
 \l_scripture_poetry_prevgraf_-
 int 3187, 3311, 3518
 \l_scripture_poetry_rightmargin_-
 tl 528, 3664
 _scripture_poetry_text_-
 right:mn ... 3188, 3188, 3207, 3638
 \l_scripture_poetry_verse_-
 right_bool 532, 534, 3475
 \l_scripture_poetry_verse_sep_-
 tl 537, 3467, 3502
 \g_scripture_post_env_indent_-
 box 1225, 1251,
 1265, 1267, 1274, 1800, 1801, 2103
 \g_scripture_prev_inner_below_-
 skip 719, 2104,
 2290, 2458, 2542, 2899, 3174, 3704
 \g_scripture_red_letter_active_-
 bool 1290, 1295, 1312
 \l_scripture_red_letter_bool ..
 286, 1293, 1309
 \l_scripture_red_letter_colour_-
 tl 289, 1296, 1302
 _scripture_red_letter_off: ...
 1307, 1307, 2097, 2253, 2274
 _scripture_red_letter_on:
 1291, 1291, 2096

\l__scripture_ref_align_str	\l__scripture_selah_text_tl
296, 297, 298, 858, 865	349, 3209, 3896
\l__scripture_ref_box	__scripture_select_language:n .
822, 845, 851, 870, 880	829, 837, 1323,
\l__scripture_ref_colour_tl	1323, 1405, 1611, 1810, 2087, 3444
302, 827, 828	__scripture_set_pilcrow_hook: .
\l__scripture_ref_font_tl	738, 1137, 1142, 1183,
__scripture_ref_format:n	1185, 1217, 1241, 1243, 1843, 1849,
309, 830, 3796	1871, 1890, 1890, 2807, 2934, 2936
\l__scripture_ref_position_str .	__scripture_setup_list_noindent:n
324, 325, 326, 946, 2226, 2243, 2321	721, 721, 2193,
\l__scripture_ref_sep_tl	2424, 2509, 2878, 3132, 3137, 3681
330, 852, 871, 877, 2263, 2268, 3797	__scripture_setup_reset_-
\l__scripture_ref_start_newline_-	defaults: 170, 669
bool 334, 337, 954, 964,	__scripture_setup_style:nnn . . .
970, 976, 1008, 1025, 1042, 1053, 1064	673, 673, 679, 688, 3725
\l__scripture_ref_start_sep_tl .	__scripture_skip_diff:NN 760, 760,
340, 892	2189, 2284, 2420, 2454, 2505, 2538,
\l__scripture_ref_tl	2874, 2895, 3147, 3170, 3677, 3700
884, 889, 891, 897, 908, 910,	\g__scripture_suppress_next_-
947, 1082, 1952, 2071, 2245, 2257,	pilcrow_bool 899, 900, 902, 1260,
2259, 2270, 2297, 2299, 2306, 2323	1270, 1428, 1597, 1598, 1602, 1648,
\l__scripture_ref_vertical_sep_-	1649, 1653, 1657, 1878, 1889, 1907,
tl 343, 938	1908, 2099, 2100, 2406, 2461, 2491,
__scripture_reference:n	2545, 2553, 2746, 2768, 2852, 2902,
843, 843, 2303, 2306	3135, 3177, 3495, 3661, 3707, 3834
\l__scripture_reference_language_-	\g__scripture_suppress_next_-
tl 312	verse_para_bool
\l__scripture_reference_language_-	737, 1201, 1216, 1773,
variant_tl 316	1854, 1855, 1872, 1886, 2101, 2806
__scripture_reference_start_-	__scripture_temp_ch 1011
inline: 882, 882, 966,	__scripture_temp_ch: 1018
972, 978, 1020, 1037, 1048, 1059, 1078	__scripture_temp_newch: . . 1028, 1035
__scripture_reference_start_-	__scripture_temp_newvs: . . 1056, 1058
newline: . . . 895, 895, 961, 965,	__scripture_temp_vs: . . . 1045, 1047
971, 977, 1009, 1026, 1043, 1054, 1066	__scripture_text_right:nn
__scripture_reference_start_-	793, 793, 2146
peek: 941, 941, 3745, 3751,	\l__scripture_text_right_sep_tl
3757, 3763, 3769, 3781, 3830, 3853,	366, 2146, 3639
3859, 3865, 3872, 3878, 3884, 3907	__scripture_vbox_set_top:Nw . . .
__scripture_reset_par_at_group_-	2906, 2906, 3009
level:	\l__scripture_verse_colour_tl . .
. . 1692, 1697, 1697, 1735, 2376, 3097	372, 1808, 1809
__scripture_reset_spacing:	__scripture_verse_compact_-
1932, 1940	output:n 1840, 1840, 3846
\l__scripture_rightmargin_tl . . .	\l__scripture_verse_first_bool .
346, 2176, 3795	376, 1813, 2157
\l__scripture_selah_font_tl	__scripture_verse_first_-
352, 3206, 3894	format:n 379, 1823, 2159
__scripture_selah_format:n	\l__scripture_verse_first_sep_tl
355, 3209, 3896	382, 1825
__scripture_selah_output:	\l__scripture_verse_font_tl 385, 1807
3203, 3203, 3635	__scripture_verse_format:n
\l__scripture_selah_sep_tl 358, 3208	388, 1828, 2159

\g__scripture_verse_int	\scripturecurrentbook 697, 702, 2081, 3716
... 369, 2075, 2085, 2121, 2135, 2137	\scripturecurrentchapter 7,
\l__scripture_verse_language_tl 393	703, 1384, 1492, 1576, 2082, 3420, 3716
\l__scripture_verse_language_-	\scripturecurrentverse . 7, 704, 1385,
variant_tl 397	1493, 1577, 1776, 2084, 3421, 3716
__scripture_verse_output:n	\scripturesetup 8, 3719
... 1480, 1559, 1774 , 1774, 1844,	\scripturestyle 8, 27, 3723
1866, 2115, 2684, 2753, 2761, 2774,	\scripturestyle* 8
2782, 2792, 2973, 2986, 3414, 3452,	\scshape 22, 268, 593
3472, 3479, 3487, 3497, 3513, 3603	\section 3, 5
\l__scripture_verse_par_start_-	\selah 8, 3635, 3891
bool	selah/font (option) 25
.. 1771 , 1778, 1779, 1783, 1794, 1814	selah/format (option) 25
\l__scripture_verse_para_bool ..	selah/sep (option) 25
... 400, 728,	selah/text (option) 25
750, 775, 923, 1213, 1271, 1412,	\selectfont 1402
1483, 1495, 1684, 1798, 1905, 1984,	\selectlanguage .. 20, 22, 1333, 1342, 1345
2326, 2430, 2515, 2688, 2834, 3155,	\setbox 2908
3604, 3607, 3653, 3812, 3840, 3911	skip commands:
__scripture_verse_para_output:n	\skip_eval:n
... 1846, 1846, 3814	... 762, 1433, 1529, 2616, 3241, 3429
\l__scripture_verse_presep_tl ..	\skip_gset_eq:NN
... 406, 1784, 1788, 2155	... 2458, 2542, 2899, 3174, 3704
\l__scripture_verse_sep_tl	\skip_gzero:N 2104
409, 1830, 2741, 2743, 2790, 3467, 3502	\skip_horizontal:N
\l__scripture_verse_show_bool 892, 1555, 1603, 1675,
... 391, 412, 1804	1833, 1913, 2263, 2268, 2671, 2763,
\l__scripture_version_delim_tl .	2784, 3091, 3279, 3447, 3489, 3510
... 418, 836, 3801	\skip_horizontal:n . 727, 811, 876,
__scripture_version_format:n ..	904, 1440, 1722, 1788, 1971, 2644,
... 421, 838, 3799	2647, 2797, 2978, 3041, 3194, 3508
\l__scripture_version_language_-	\skip_if_eq:nnTF 1429,
tl 424	1525, 1784, 2155, 2612, 3237, 3425
\l__scripture_version_language_-	\skip_new:N 719
variant_tl 428	\skip_set:Nn 1921, 2183,
\l__scripture_version_position_-	2187, 2282, 2418, 2452, 2503, 2536,
str 434, 435, 436, 886, 905, 1083, 2246	2872, 2893, 3145, 3168, 3675, 3698
\l__scripture_version_tl	\skip_set_eq:NN
... 415, 847, 889,	... 1922, 2412, 2497, 2866, 3139
908, 1084, 2247, 2261, 2264, 2270, 2301	\skip_vertical:N 857, 862
\l__scripture_X_char_tl 63,	\skip_vertical:n ... 1948, 2290, 3089
167, 1016, 1033, 1378, 2111, 2830, 3630	\skip_zero:N 816, 1579, 1923,
__scripture_zero_chap_par_-	1924, 2182, 2185, 2275, 2280, 2413,
prevgraf_at_group_level:	2416, 2450, 2498, 2501, 2534, 2867,
... 1758, 1758, 1768, 3234	2870, 2891, 2980, 3016, 3140, 3143,
scripture/chap/after (hook) 12, 47	3166, 3199, 3670, 3671, 3673, 3696
scripture/chap/before (hook) 12, 47	\c_zero_skip 33, 48, 64, 545,
scripture/heading (mark) 13, 47	550, 555, 1429, 1525, 1555, 1784,
scripture/pilcrow (hook) 47	1833, 1913, 2155, 2612, 3237, 3425
scripture/poetry/para/after (hook) . 102	\small 19, 215, 577
scripture/poetry/pilcrow (hook) 47	\smash 16, 1536
scripture/verse (mark) 13, 47	\spacefactor 2948, 2971
scripture/verse/after (hook) 12, 47	str commands:
scripture/verse/before (hook) 12, 47	\str_case:nn 70, 146, 189

`\str_if_eq:nnTF`
 741, 752, 827, 858, 865, 886,
 903, 905, 1073, 1083, 1152, 1283,
 1297, 1313, 1403, 1414, 1507, 1562,
 1583, 1599, 1608, 1656, 1688, 1808,
 1962, 2165, 2243, 2246, 2321, 3442
`\str_if_eq_p:nn`
 125, 796, 797, 912, 913,
 926, 930, 946, 950, 951, 993, 998,
 1117, 1122, 1176, 1620, 1621, 1622,
 1818, 2204, 2208, 2212, 2221, 2225
`\str_if_exist:N` 296, 324, 434
`\str_new:N` 297, 325, 435, 1951
`\str_set:Nn` 2072, 2387,
 2472, 2819, 2993, 3106, 3624, 3827,
 3852, 3858, 3864, 3870, 3877, 3883
`\str_set_eq:NN` 298, 326, 436
`\strut` 801, 916, 2748, 2769, 2949, 3469
`style (option)` 25

T

T_EX and L^AT_EX 2_ε commands:

`\@afterheading` 937, 1631, 1999
`\@beginparpenalty` 2092
`\@clubpenalty` 1256
`\@doendpe` 60, 3817
`\@endparpenalty` 2093
`\@ifpackageloaded` 1331, 1336
`\@lowpenalty` 2091, 2092, 2093
`\@nolnerr` 2437, 2522, 3823
`\@outerparskip`
 2279, 2285, 2449, 2455, 2533, 2539,
 2890, 2896, 3165, 3171, 3695, 3701
`\@par` 2236, 3816
`\@restorepar` 1255
`\@topsepadd`
 2280, 2282, 2450, 2452, 2458,
 2534, 2536, 2542, 2891, 2893, 2899,
 3166, 3168, 3174, 3696, 3698, 3704
`\@totalleftmargin`
 96, 1449, 1461, 1471,
 1571, 1733, 1862, 1880, 1944, 1966,
 2278, 2308, 2345, 2348, 2358, 2447,
 2463, 2531, 2547, 2589, 2599, 2607,
 2718, 2727, 2729, 2731, 2733, 2904,
 3013, 3014, 3179, 3295, 3300, 3341,
 3351, 3388, 3398, 3520, 3523, 3713
`\f@size` 1391
`\textright` 8, 2145, 3636, 3900
`textright/sep (option)` 25
`\textscripture` 9, 3727
`\textsuperscript` 26, 390, 624

tl commands:

`\c_space_tl` 19, 21, 248,
 420, 587, 633, 2972, 3895, 3897, 3903
`\tl_clear:N` 1082, 1084, 2245, 2247
`\tl_gclear:N` 2076, 2323
`\tl_gset:Nn` 683,
 1384, 1385, 1492, 1493, 1576, 1577,
 1776, 2081, 2082, 2084, 3420, 3421
`\tl_if_empty:N`TF
 697, 884, 897, 1329, 1340, 2261, 2301
`\tl_if_empty:n`TF 833, 835
`\tl_if_empty_p:N` 947, 2259, 2299
`\tl_if_eq:NnTF` 1514, 1520
`\tl_if_eq:nnTF` 1786
`\tl_if_exist:N`TF ... 68, 144, 187, 677
`\tl_if_no`value:nTF .. 1014, 1031,
 2070, 2077, 2109, 2388, 2473, 2820,
 2828, 2994, 3107, 3628, 3729, 3774
`\tl_if_no`value_p:n 2257, 2297
`\tl_new:N` 69, 145,
 188, 542, 678, 1952, 3716, 3717, 3718
`\tl_set:Nn`
 74, 79, 150, 155, 160, 193, 198,
 203, 1016, 1033, 2071, 2111, 2161,
 2830, 3630, 3792, 3793, 3794, 3795
`\tl_set_eq:NN` ... 1325, 1327, 2741,
 2742, 2790, 3467, 3502, 3797, 3800
`\tl_use:N` . 680, 3071, 3072, 3078, 3079
`\l_tmpa_tl`
 2741, 2790, 3068, 3071, 3076, 3078
`\l_tmpb_tl` 3068, 3072, 3076, 3079

token commands:

`\c_space_token` 1637, 2371, 2924
`\token_if_active_p:N` 1180, 1238
`\token_if_eq_meaning:N`TF 1080
`\token_if_eq_meaning_p:NN`
 733, 734, 960, 962, 968,
 974, 985, 1006, 1023, 1040, 1051,
 1095, 1097, 1099, 1101, 1109, 1130,
 1132, 1134, 1139, 1174, 1182, 1184,
 1199, 1200, 1210, 1211, 1240, 1242,
 2803, 2804, 2933, 2935, 3549, 3557,
 3565, 3567, 3569, 3579, 3586, 3595
`\token_if_letter_p:N` 1178, 1236, 2931
`\token_to_str:N` 19, 21, 27
`\TopMark` 13
`\topsep` 83,
 2185, 2187, 2416, 2418, 2501, 2503,
 2870, 2872, 3143, 3145, 3673, 3675

U

`\unpenalty` 2957, 3028

<code>\unskip</code>	799, 808, 2252, 2378, 2436, 2521, 2947, 2955, 2956, 3026, 3027, 3191, 3810, 3822, 3887, 3902	
<code>\usepackage</code>	13	
V		
<code>\vbadness</code>	3059	
<code>\vbox</code>	3	
vbox commands:		
<code>\vbox_set_end:</code>	3047	
<code>\vbox_set_split_to_ht:NNn</code>	3060	
<code>\vbox_set_top:Nn</code>	3064	
<code>\vbox_set_top:Nw</code>	95	
<code>\vbox_unpack_drop:N</code>	3065	
<code>verse</code> (option)	25	
<code>verse/color</code> (option)	26	
<code>verse/colour</code> (option)	26	
<code>verse/first</code> (option)	26	
<code>verse/firstformat</code> (option)	26	
<code>verse/firstsep</code> (option)	26	
<code>verse/font</code> (option)	26	
<code>verse/format</code> (option)	26	
<code>verse/hide</code> (option)	26	
<code>verse/language</code> (option)	27	
<code>verse/language/variant</code> (option)	27	
<code>verse/para</code> (option)	27	
<code>verse/presep</code> (option)	28	
<code>verse/sep</code> (option)	28	
<code>verse/show</code> (option)	28	
<code>version</code> (option)	28	
<code>version/delim</code> (option)	28	
<code>version/format</code> (option)	28	
<code>version/language</code> (option)	28	
<code>version/language/variant</code> (option) ...	28	
<code>version/position</code> (option)	29	
<code>\vs</code>	9, 48, 733, 1040, 1045, 1046, 1047, 1134, 1182, 1199, 1210, 1240, 1568, 1645, 1651, 1658, 2115, 2137, 2803, 2835, 2836, 2933, 3245, 3313, 3433, 3459, 3497, 3513, 3540, 3544, 3572, 3579, 3582, 3589, 3597, 3601, 3603, 3634, 3814, 3846	
<code>\vtop</code>	2908	