

The `scripture` package

David Purton*

2026-08-23 v2.5

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	30
2.5.3	Options for the <code>hanging</code> environment	30
2.5.4	Options for the <code>midparachap</code> environment	32
2.5.5	Options for the <code>narrow</code> environment	32
2.5.6	Options for the <code>poetry</code> environment	33

*Email: dcpurton@marshwiggles.net

3	Implementation	34
3.1	Messages	34
3.2	Options	35
3.3	Hooks	49
3.4	Marks	49
3.5	The main <code>scripture</code> environment	50
3.5.1	Formatting and printing quotation references	53
3.5.2	Red lettering	62
3.5.3	Selecting language	63
3.5.4	Formatting and printing chapter numbers	64
3.5.5	Formatting and printing verse numbers	74
3.5.6	Printing pilcrows	76
3.5.7	Calculating the length of the final line in a paragraph	77
3.5.8	The <code>scripture</code> environment	78
3.6	Scripture internal environments	86
3.6.1	The <code>center</code> environment	88
3.6.2	The <code>flushright</code> environment	90
3.6.3	The <code>hanging</code> environment	91
3.6.4	The <code>midparachap</code> environment	99
3.6.5	The <code>narrow</code> environment	103
3.6.6	The <code>poetry</code> environment	105
3.7	Public L ^A T _E X Interface	116
	 Change History	 122
	 Index	 124

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)**

`\canonicalheading` `\canonicalheading{<canonical heading>}`

New: 2026-08-23 The `\canonicalheading` function inserts a canonical heading used at the beginning of many psalms.

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

- `canonicalheading/aboveskip`
- `canonicalheading/afterindent`
- `canonicalheading/align`
- `canonicalheading/belowskip`
- `canonicalheading/font`
- `canonicalheading/format`

```
\begin{scripture}[] [chapter/para,
  chapter/format=PSALM-#1]
\ch{3}
\canonicalheading{A psalm of David. When he fled
  from his son Absalom.}
\begin{poetry}
  \vs{1}\name{Lord}, how many are my foes!
  How many rise up against me!
\end{poetry}
\end{scripture}
```

PSALM 3

A psalm of David. When he fled from his son Absalom.

¹LORD, how many are my foes!
How many rise up against me!

<code>\ch</code>	<code>\ch[⟨letter⟩]{⟨chapter number⟩}</code>
<code>\ch*</code>	<code>\ch*[⟨letter⟩]{⟨chapter number⟩}</code>
<code>\newch</code>	<code>\newch[⟨letter⟩]</code>
<code>\newch*</code>	<code>\newch*[⟨letter⟩]</code>

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 `scirpture` 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:
²I 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:
²I am the LORD your God, who
brought you out of Egypt, out of the
land of slavery. **Exodus 20:1–2**

<code>\extraskip</code>	<code>\extraskip</code>
-------------------------	-------------------------

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 Colos-
sae, the faithful brothers and sisters
in Christ:

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

<code>\heading</code>	<code>\heading{<section heading>}</code>
-----------------------	--

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)

`\nofirstverse` `\nofirstverse`

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.

`\nohang` `\nohang`

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.

`\scripturesetup` `\scripturesetup{<options>}`

Set default format for quotations. See section 2.5 for supported formatting `<options>`.

`\scripturestyle` `\scripturestyle{<name>}{<options>}`
`\scripturestyle*` `\scripturestyle*{<name>}{<options>}`

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.

`\selah` `\selah`

New: 2022-07-29

In a `poetry` environment, this function inserts *Selah* 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.

```

\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}

```

²Many are saying of me,
“God will not deliver him.” *Selah*
Psalm 3:2 (NIV84)

`\textright` `\textright{<text>}`

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.

<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>	⁵ ‘Why do you bring charges against me? You have all rebelled against me,’ declares the LORD. Jeremiah 2:29
--	--

`\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: <pre>\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.}</pre>	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}[\langle options \rangle]`
`\end{center}`

Environment for typesetting centred sections of scripture quotations.
 See section 2.5.1 for supported formatting `\langle options \rangle`.

<pre>\begin{scripture}[Daniel 5:25] \vs{25}`This is the inscription that was written: \begin{center} \textsc{mene, mene, tekel, parsin} \end{center} \end{scripture}</pre>	²⁵ “This is the inscription that was written: MENE, MENE, TEKEL, PARSIN Daniel 5:25
--	---

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

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

<pre>\begin{scripture}[Isaiah 40:5] \begin{poetry}[belowskip = 0pt] \vs{5}And the glory of the \name{Lord}\ldots and all people will see it together. \end{poetry} \begin{flushright}[aboveskip = 0pt] For the mouth of the \name{Lord} \\\ has spoken.' \end{flushright} \end{scripture}</pre>	⁵ And 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]`
`\end{hanging}`

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

<pre>\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}</pre>	³ And God said, ‘Let there be light,’ and there was light. ⁴ God saw that the light was good, and he separated the light from the darkness. ⁵ God 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 $\langle options \rangle$.

```
\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.

11 **1 Corinthians 10:31–11:1**

```
\narrow \begin{narrow}[\langle options \rangle]
\text{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 $\langle options \rangle$.

```
\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}
\text{‘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}[\langle options \rangle]
\text{first half of verse}
\text{second half of verse}

\text{first half of verse}
\text{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 per- ish 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 = <font>` Default: Empty

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

added/format	<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> .	
after	<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.	
before	<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.	
belowskip	<code>belowskip = $\langle skip \rangle$</code>	Default: <code>Opt</code>
	$\langle skip \rangle$ inserted below the <code>scripture</code> environment.	
book	<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> .	
canonicalheading/aboveskip	<code>canonicalheading/aboveskip = $\langle skip \rangle$</code>	Default: <code>\medskipamount</code>
New: 2026-08-23	$\langle skip \rangle$ inserted above a <code>\canonicalheading</code> .	
canonicalheading/afterindent	<code>canonicalheading/afterindent = $\langle boolean \rangle$</code>	Default: <code>false</code>
New: 2026-08-23	Whether to indent the first paragraph following a <code>\canonicalheading</code> . The value of $\langle boolean \rangle$ is optional and assumed to be <code>true</code> if not present.	
canonicalheading/align	<code>canonicalheading/align = left right center</code>	Default: <code>center</code>
New: 2026-08-23	Set the alignment of a <code>\canonicalheading</code> .	
canonicalheading/belowskip	<code>canonicalheading/belowskip = $\langle skip \rangle$</code>	Default: <code>\medskipamount</code>
New: 2026-08-23	$\langle skip \rangle$ inserted below a <code>\canonicalheading</code> .	
canonicalheading/font	<code>canonicalheading/font = $\langle font \rangle$</code>	Default: <code>\footnotesize</code>
New: 2026-08-23	$\langle font \rangle$ used for a <code>\canonicalheading</code> .	
canonicalheading/format	<code>canonicalheading/format = $\langle macro \rangle$</code>	Default: <code>#1</code>
New: 2026-08-23	$\langle macro \rangle$ used to format a <code>\canonicalheading</code> . #1 contains the text of the heading.	
chapter	<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> .	

chapter/aboveskip	chapter/aboveskip = $\langle skip \rangle$ Default: Opt	
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> .	
chapter/align	chapter/align = left right	Default: left
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> .	
chapter/colour	chapter/colour = $\langle colour\ expression \rangle$	Default: .
chapter/color	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		
chapter/drop	chapter/drop = $\langle boolean \rangle$	Default: true
	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> .	
chapter/font	chapter/font = $\langle font \rangle$	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 <code>scripture</code> 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 <code>true</code> if not present. This option is the opposite of <code>chapter/show</code> . Note that the <code>scripture/chap/before</code> and <code>scripture/chap/after</code> 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> 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. Romans 5:1-2		
chapter/language	chapter/language = $\langle language\ name \rangle$	Default: Empty
New: 2026-06-03	<code>babel</code> or <code>polyglossia</code> $\langle language\ name \rangle$ to be used for chapter numbers in case you want to use the main document language instead of the Bible text language.	

<code>chapter/language/variant</code>	<code>chapter/language/variant = <language variant name></code>	Default: <i>Empty</i>
---------------------------------------	---	-----------------------

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

<code>chapter/nodrop</code>	<code>chapter/nodrop = <boolean></code>	Default: <i>false</i>
-----------------------------	---	-----------------------

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 `<boolean>` is optional and assumed to be **true** if not present. This option is the opposite of `chapter/drop`.

```
\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}
```

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 noth-
ing was made that has been made.

John 1:1-3

<code>chapter/para</code>	<code>chapter/para = <boolean></code>	Default: <i>false</i>
---------------------------	---	-----------------------

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 `<boolean>` 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)

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

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.

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

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.

<code>chapter/para/indent</code>	<code>chapter/para/indent = <boolean></code>	Default: <i>true</i>
----------------------------------	--	----------------------

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.

<code>chapter/presep</code>	<code>chapter/presep = <skip></code>	Default: <i>0em</i>
-----------------------------	--	---------------------

New: 2026-08-20 `<skip>` inserted before `\ch` when the `compact` option is in use.

<code>chapter/sep</code>	<code>chapter/sep = <dimen></code>	Default: 0.5em
--------------------------	--	----------------

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

<code>chapter/show</code>	<code>chapter/show = <boolean></code>	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`.

<code>chapter/showverse</code>	<code>chapter/showverse = <boolean></code>	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}
```

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

<code>chapter/smash</code>	<code>chapter/smash = <boolean></code>	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: bottom
-----------------------------	--	-----------------

New: 2024-12-30 Sets how a chapter will align vertically when `chapter/nodrop` 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 ὁδὸν 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· ² καθὼς γέγρα-
πται ἐν τῷ Ἡσαΐα τῷ προφή-
τῃ· ἰδοὺ ἀποστέλλω τὸν ἄγγελόν
μου πρὸ προσώπου σου, ὃς κα-
τασκευάσει τὴν ὁδὸν σου. ³ φω-
νὴ βοῶντος ἐν τῇ ἐρήμῳ ἐτοι-
μάσατε τὴν ὁδὸν κυρίου, εὐθείας
ποιεῖτε τὰς τρίβους αὐτοῦ.
Κατά Μάρκον 1:1–3 (THGNT)

chapter/xchar	<code>chapter/xchar = $\langle$character$\rangle$</code>	Default: X
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> .	
colour	<code>colour = $\langle$colour expression$\rangle$</code>	Default: .
color	The <code>\color</code> $\langle$ colour expression $\rangle$ to be used for the scripture environment. By default the current colour is inherited. color is an alias for colour .	
New: 2022-08-31		
compact	<code>compact = $\langle$boolean$\rangle$</code>	Default: false
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 chapter/nodrop and all options relating to inner environments are ignored. The value of $\langle$ boolean $\rangle$ is optional and assumed to be true if not present. Warning: This option redefines <code>\par</code> to <code>\relax</code> inside the scripture environment. If you have anything other than expected macros and environments inside your scripture environment T _E X 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 defaults

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

extraskip extraskip = $\langle skip \rangle$

Default: `\medskipamount`

$\langle skip \rangle$ used for the `\extraskip` function.

font font = $\langle font \rangle$

Default: *Empty*

$\langle font \rangle$ used for the `scripture` environment.

heading/aboveskip heading/aboveskip = $\langle skip \rangle$

Default: `\bigskipamount`

New: 2024-11-19 $\langle skip \rangle$ inserted above a `\heading`.

heading/afterindent heading/afterindent = $\langle boolean \rangle$

Default: `false`

New: 2024-11-19 Whether to indent the first paragraph following a `\heading`. The value of $\langle boolean \rangle$ is optional and assumed to be `true` if not present.

heading/align heading/align = `left` | `right` | `center`

Default: `left`

New: 2024-11-19 Set the alignment of a `\heading`.

heading/belowskip heading/belowskip = $\langle skip \rangle$

Default: `\medskipamount`

New: 2024-11-19 $\langle skip \rangle$ inserted below a `\heading`.

heading/font heading/font = $\langle font \rangle$

Default: `\small\itshape`

New: 2024-11-19 $\langle font \rangle$ used for a `\heading`.

heading/format heading/format = $\langle macro \rangle$

Default: `#1`

New: 2024-11-19 $\langle macro \rangle$ used to format a `\heading`. `#1` contains the text of the heading.

heading/hide	<code>heading/hide = <boolean></code>	Default: <code>false</code>
New: 2024-11-19	Whether to hide <code>\heading</code> functions. The value of <code><boolean></code> is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>heading/show</code> .	
heading/show	<code>heading/show = <boolean></code>	Default: <code>true</code>
New: 2024-11-19	Whether to show <code>\heading</code> functions. The value of <code><boolean></code> is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>heading/hide</code> .	
indent	<code>indent = <boolean></code>	Default: <code>true</code>
	Whether to indent the first line of a <code>scripture</code> environment or not by <code>parindent</code> . The value of <code><boolean></code> is optional and assumed to be <code>true</code> if not present. This option is the opposite of <code>noindent</code> .	
inline	<code>inline = <boolean></code>	Default: <code>false</code>
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 <code><reference></code> inline. The reference format can be controlled using the following options:	
	• <code>inline/reference/format</code> • <code>inline/reference/sep</code> • <code>inline/version/delim</code> • <code>inline/version/format</code>	
	Other <code><reference></code> and <code><version></code> 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.	
inline/begin	<code>inline/begin = <text></code>	Default: <code>`</code>
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.	
inline/end	<code>inline/end = <text></code>	Default: <code>'</code>
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 <code><reference></code> (if specified).	
inline/font	<code>inline/font = </code>	Default: <code>Empty</code>
New: 2026-05-31	<code></code> used for <code>\textscripture</code> or a <code>scripture</code> environment with the <code>inline</code> option set.	
inline/reference/format	<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).	

<code>inline/reference/sep</code>	<code>inline/reference/sep = <dimen></code>	Default: 0.5em
New: 2023-12-27	The <dimen> between the end of an inline scripture quotation and its <reference>.	
<code>inline/version/delim</code>	<code>inline/version/delim = <delim></code>	Default: { }
New: 2023-12-27	<delim> to use between the <reference> and the <version> of an inline scripture quotation.	
<code>inline/version/format</code>	<code>inline/version/format = <macro></code>	Default: #1
New: 2023-12-27	<macro> used to format the <version> for an inline scripture quotation. #1 contains the text of the <version>.	

<code>language</code>	<code>language = <language name></code>	Default: Empty
New: 2022-11-02	<code>babel</code> or <code>polyglossia</code> <language name> to be used for the scripture 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][version=NMV,
  language=persian]
  ch{56}
  \begin{poetry}
    «انصاف را نگاه داشته،
    عدالت را به عمل آورید،
    زیرا نجات من نزدیک شده،
    و عدالت من به زودی منکشف خواهد شد.

    ۲ خوشا به حال آن که چنین کند،
    و کسی که بدین متمسک گردد؛
    که شَبَات را نگاه داشته، آن را بی حرمت نسازد،
    و دست خویش از هر عمل بد باز دارد.»
    \end{poetry}
  \end{scripture}
\end{document}
```

خداوند چنین می فرماید:

۵۶

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

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)

language/variant	language/variant = $\langle$ language variant name $\rangle$	Default: Empty
------------------	--	----------------

New: 2022-11-02 polyglossia $\langle$ language variant name $\rangle$ to be used along with the specified **language** for the **scripture** environment. This inserts `\selectlanguage[ $\langle$ language variant name $\rangle$ ]{ $\langle$ language name $\rangle$ }` at the start of the environment. Make sure you have polyglossia loaded and correctly set up for your specified language.

leftmargin	leftmargin = $\langle$ dimen $\rangle$	Default: Opt
------------	--	--------------

$\langle$ dimen $\rangle$ used for the left margin of the **scripture** environment.

mark/verse/format	mark/verse/format = $\langle$ macro $\rangle$	Default: #1 #2:#3
-------------------	---	-------------------

New: 2026-05-20 $\langle$ macro $\rangle$ used to format the **scripture/verse** mark. The arguments are:

#1 Current book name.

#2 Current chapter.

#3 Current verse.

name/font	name/font = $\langle$ font $\rangle$	Default: \scshape
-----------	--------------------------------------	-------------------

New: 2022-09-01 $\langle$ font $\rangle$ used for the **\name** function.

name/format	name/format = $\langle$ macro $\rangle$	Default: #1
-------------	---	-------------

New: 2022-09-01 $\langle$ macro $\rangle$ used to format the divine name using the **\name** function. #1 contains the text of the argument to **\name**.

noindent `noindent = <boolean>` Default: `false`

Whether to indent the first line of a `scripture` environment or not by `parindent`. The value of `<boolean>` 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>	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. 1 Peter 3:18
---	---

parindent `parindent = <dimen>` Default: `\parindent`

`<dimen>` to use when indenting the first line of a paragraph in the `scripture` environment.

parskip `parskip = <skip>` Default: `\parskip`

`<skip>` to use between paragraphs in the `scripture` environment.

pilcrow `pilcrow = <boolean>` Default: `false`

New: 2024-01-18 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 `<boolean>` 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.

<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>	CHAPTER 5 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
---	---

pilcrow/sep `pilcrow/sep = <dimen>` Default: `0.25em`

New: 2024-01-18 `<dimen>` used to separate a pilcrow (¶) from the following text.

redletter	<code>redletter = <boolean></code>	Default: <code>false</code>
New: 2022-08-31	Whether to enable highlighting of Jesus' words in red. The value of <code><boolean></code> is optional and assumed to be <code>true</code> if not present. The text needs to be marked up with <code>\redletteron</code> and <code>\redletteroff</code> for this option to have any effect.	
redletter/colour	<code>redletter/colour = <colour expression></code>	Default: <code>red!80!black</code>
redletter/color	The <code>!3color <colour expression></code> to be used for the words of Jesus when the <code>redletter</code> option is <code>true</code> . <code>\redletteron</code> and <code>\redletteroff</code> must be used to mark up the text. <code>redletter/color</code> is an alias for <code>redletter/colour</code> .	
New: 2022-08-31		
reference/align	<code>reference/align = left right</code>	Default: <code>right</code>
New: 2023-01-08	Set the alignment of the reference. Note that in neither case will the reference break across lines.	
reference/colour	<code>reference/colour = <colour expression></code>	Default: <code>.</code>
reference/color	The <code>!3color <colour expression></code> to be used for the reference and version. By default the current colour is inherited. <code>reference/color</code> is an alias for <code>reference/colour</code> .	
New: 2022-09-07		
reference/font	<code>reference/font = </code>	Default: <code>\bfseries</code>
	<code></code> used when printing the <code><reference></code> for a <code>scripture</code> quotation.	
reference/format	<code>reference/format = <macro></code>	Default: <code>#1</code>
	<code><macro></code> used to format the <code><reference></code> for a <code>scripture</code> quotation. <code>#1</code> contains the text of the <code><reference></code> and <code><version></code> (if specified).	
reference/language	<code>reference/language = <language name></code>	Default: <code>Empty</code>
New: 2026-06-03	<code>babel</code> or <code>polyglossia <language name></code> to be used for the reference in case you want to use the main document language instead of the Bible text language.	
reference/language/variant	<code>reference/language/variant = <language variant name></code>	Default: <code>Empty</code>
New: 2026-06-03	<code>polyglossia <language variant name></code> to be used for the reference in case you want to use the main document language variant instead of the Bible text language variant.	
reference/newline	<code>reference/newline</code>	
New: 2023-01-08	Always push the <code><reference></code> to a newline even if there is space for it to fit on the last line of the quotation. The option sets <code>reference/sep</code> to <code>\linewidth</code> so there is never room for the reference on the final line of the quotation.	
reference/position	<code>reference/position = start end</code>	Default: <code>end</code>
New: 2024-01-10	Set the position of the reference at either the start or end of the quotation. If <code>start</code> is specified, then the reference will be set inline unless the quotation begins with an inner environment. Any <code>\vs</code> or <code>\ch</code> is suppressed at the start of the quotation in the main <code>scripture</code> environment.	


```
\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}
```

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

reference/sep reference/sep = $\langle \text{dimen} \rangle$ Default: 2em

The minimum $\langle \text{dimen} \rangle$ allowed between the end of a **scripture** quotation and its $\langle \text{reference} \rangle$ before the $\langle \text{reference} \rangle$ is pushed to the following line.

reference/start/inline reference/start/inline = $\langle \text{boolean} \rangle$ Default: true

New: 2024-01-10

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 $\langle \text{boolean} \rangle$ is optional and assumed to be **true** if not present. This option is the opposite of **reference/start/newline**.

reference/start/newline reference/start/newline = $\langle \text{boolean} \rangle$ Default: false

New: 2024-01-10

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 $\langle \text{boolean} \rangle$ is optional and assumed to be **true** if not present. This option is the opposite of **reference/start/inline**.

```
\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	<code>selah/format = <macro></code>	Default: #1
	<code><macro></code> used to format the <code>\selah</code> in a poetry environment. #1 contains the value of <code>selah/text</code> .	
selah/sep	<code>selah/sep = <dimen></code>	Default: 1em
	The minimum <code><dimen></code> allowed between the end of a poetry line and <i>Selah</i> before the <i>Selah</i> is pushed to the following line.	
selah/text	<code>selah/text = <text></code>	Default: <i>Selah</i>
	<code><text></code> used to for <code>\selah</code> in a poetry environment.	
style	<code>style = <name></code>	
New: 2023-12-27	Set options previously saved with <code>\scripturestyle</code> . An error occurs if the style isn't defined.	
textright/sep	<code>textright/sep = <dimen></code>	Default: 1em
New: 2022-08-30	The minimum <code><dimen></code> allowed between the end of a line and text aligned to the right with <code>\textright</code> before the right aligned text is pushed to the following line.	
verse	<code>verse = <integer></code>	Default: 1
New: 2025-03-02	Initialise the chapter number when using automatic verse numbering with <code>\news</code> . See also book and chapter .	
verse/colour	<code>verse/colour = <colour expression></code>	Default: .
verse/color	The <code>l3color</code> <code><colour expression></code> 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	<code>verse/first = <boolean></code>	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 <code>\nofirstverse</code> 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 <code><boolean></code> 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 = $\langle macro \rangle$	Default: #1
--------------------------	---	-------------

New: 2022-11-13	$\langle macro \rangle$ 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 = $\langle dimen \rangle$	Default: 0.5em
-----------------------	--	----------------

New: 2022-11-13	$\langle dimen \rangle$ used to separate the first \vs in a paragraph from the following text. Only used if verse/first is true.
--------------------------------	--

verse/font	verse/font = $\langle font \rangle$	Default: Empty
-------------------	-------------------------------------	----------------

$\langle font \rangle$ used for the **\vs** function.

verse/format	verse/format = $\langle macro \rangle$	Default: $\text{\#1}$
---------------------	--	---

$\langle macro \rangle$ used to format the **\vs** function. #1 contains the argument of **\vs**.

verse/hide	verse/hide = $\langle boolean \rangle$	Default: false
-------------------	--	----------------

New: 2022-08-15	Whether verses 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 verse/show . Note that the scripture/verse/before and scripture/verse/after hooks are still executed when verses are hidden.
--------------------------------	--

<pre> \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> </pre>	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 = $\langle language name \rangle$	Default: Empty
-----------------------	--	----------------

New: 2026-06-03	babel or polyglossia $\langle language name \rangle$ 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 = $\langle language variant name \rangle$	Default: Empty
-------------------------------	--	----------------

New: 2026-06-03	polyglossia $\langle language variant name \rangle$ 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 = $\langle boolean \rangle$	Default: false
-------------------	--	----------------

New: 2024-01-13	Whether to set each verse on its own line. The value of $\langle boolean \rangle$ 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.

<u>version/delim</u>	version/delim = $\langle delim \rangle$	Default: { }
	$\langle delim \rangle$ to use between the $\langle reference \rangle$ and the $\langle version \rangle$ of a scripture quotation.	

<u>version/format</u>	version/format = $\langle macro \rangle$	Default: (#1)
	$\langle macro \rangle$ used to format the $\langle version \rangle$ for a scripture quotation. #1 contains the text of the $\langle version \rangle$.	

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

<u>version/language/variant</u>	version/language/variant = $\langle language variant name \rangle$	Default: Empty
New: 2026-06-03	polyglossia $\langle language variant name \rangle$ 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: 2024-01-20	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.	

<pre>\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}</pre>	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: \medskipamount
	$\langle skip \rangle$ inserted above the center environment.	

<u>center/belowskip</u>	center/belowskip = $\langle skip \rangle$	Default: \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 = <code><skip></code> <code><skip></code> inserted above the <code>flushright</code> environment.	Default: <code>\medskipamount</code>
<u>flushright/belowskip</u>	flushright/belowskip = <code><skip></code> <code><skip></code> inserted below the <code>flushright</code> environment.	Default: <code>\medskipamount</code>
<u>flushright/leftmargin</u>	flushright/leftmargin = <code><dimen></code> <code><dimen></code> used for the left margin of the <code>flushright</code> environment.	Default: <code>Opt</code>
<u>flushright/rightmargin</u>	flushright/rightmargin = <code><dimen></code> <code><dimen></code> used for the right margin of the <code>flushright</code> environment.	Default: <code>Opt</code>

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.

<u>hanging/aboveskip</u>	hanging/aboveskip = <code><skip></code> <code><skip></code> inserted above the <code>hanging</code> environment.	Default: <code>\medskipamount</code>
<u>hanging/belowskip</u>	hanging/belowskip = <code><skip></code> <code><skip></code> inserted below the <code>hanging</code> environment.	Default: <code>\medskipamount</code>
<u>hanging/chapter/indent</u>	hanging/chapter/indent = <code><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>

New: 2024-12-31

```
\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

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

```

\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 ἀκοή αὐτοῦ εὐθὺς πανταχοῦ
εἰς ὅλην τὴν περίχωρον τῆς Γαλιλαίας.

\vs{29}Καὶ εὐθὺς ἐκ τῆς συναγωγῆς ἐξεληθόντες
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 ἀκοή αὐτοῦ εὐθὺς πανταχοῦ εἰς ὅλην τὴν περίχωρον τῆς Γαλιλαίας.

29 Καὶ εὐθὺς ἐκ τῆς συναγωγῆς ἐξεληθόντες 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: <code>false</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/showverse`. See also `chapter/showverse`.

<pre>\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}</pre>	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.

narrow/rightmargin	narrow/rightmargin = $\langle dimen \rangle$ $\langle dimen \rangle$ used for the right margin of the narrow environment.	Default: 1em
--------------------------------	---	--------------

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.

poetry/aboveskip	poetry/aboveskip = $\langle skip \rangle$ $\langle skip \rangle$ inserted above the poetry environment.	Default: \medskipamount
------------------------------	---	-------------------------

poetry/belowskip	poetry/belowskip = $\langle skip \rangle$ $\langle skip \rangle$ inserted below the poetry environment.	Default: \medskipamount
------------------------------	---	-------------------------

poetry/bigindent	poetry/bigindent = $\langle dimen \rangle$ $\langle dimen \rangle$ 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: 4em
------------------------------	---	--------------

poetry/indent	poetry/indent = $\langle dimen \rangle$ $\langle dimen \rangle$ used for the second half of a line of poetry in a poetry environment.	Default: 1em
---------------------------	---	--------------

poetry/leftmargin	poetry/leftmargin = $\langle dimen \rangle$ $\langle dimen \rangle$ used for the left margin of the poetry environment.	Default: 1em
-------------------------------	---	--------------

poetry/rightmargin	poetry/rightmargin = $\langle dimen \rangle$ $\langle dimen \rangle$ used for the right margin of the poetry environment.	Default: Opt
--------------------------------	---	--------------

poetry/secondhalf	poetry/secondhalf = $\langle boolean \rangle$ set the verse line of the poetry environment as a second half line.	Default: false
-------------------------------	---	----------------

poetry/verse/left	poetry/verse/left = $\langle boolean \rangle$ Align verse numbers in poetry environments at the left margin of the scripture environment. This option is the opposite of poetry/verse/right .	Default: false
-------------------------------	---	----------------

```

\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

poetry/verse/right `poetry/verse/right = <boolean>` Default: true

New: 2022-07-31 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**.

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

New: 2022-07-31 `<dimen>` used to separate a `\vs` from the following text in a **poetry** environment. This is only used for verses at the start of a line.

3 Implementation

```

1 %<*package>
2 %<@@=scripture>
3 \NeedsTeXFormat{LaTeX2e}[2022-11-01]
4 \ProvidesExplPackage{scripture}{2026-08-23}{2.5}
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~tried~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   , canonicalheading / aboveskip .tl_set:N =
53     \l__scripture_canonicalheading_aboveskip_tl
54   , canonicalheading / aboveskip .value_required:n = true
55   , canonicalheading / aboveskip .initial:n = \medskipamount
56   , canonicalheading / afterindent .bool_set:N =
57     \l__scripture_canonicalheading_afterindent_bool
```

```

58 , canonicalheading / afterindent .default:n = true
59 , canonicalheading / afterindent .initial:n = false
60 , canonicalheading / align .choices:nn =
61   { left, right, center }
62   {
63     \tl_if_exist:NF \l__scripture_canonicalheading_align_tl
64     { \tl_new:N \l__scripture_canonicalheading_align_tl }
65     \str_case:Vn \l_keys_choice_str
66     {
67       { left }
68       {
69         \tl_set:Nn \l__scripture_canonicalheading_align_tl
70           { \raggedright }
71       }
72       { right }
73       {
74         \tl_set:Nn \l__scripture_canonicalheading_align_tl
75           { \raggedleft }
76       }
77       { center }
78       {
79         \tl_set:Nn \l__scripture_canonicalheading_align_tl
80           { \centering }
81       }
82     }
83   }
84 , canonicalheading / align .value_required:n = true
85 , canonicalheading / align .initial:n = center
86 , canonicalheading / belowskip .tl_set:N =
87   \l__scripture_canonicalheading_belowskip_tl
88 , canonicalheading / belowskip .value_required:n = true
89 , canonicalheading / belowskip .initial:n = \medskipamount
90 , canonicalheading / font .tl_set:N = \l__scripture_canonicalheading_font_tl
91 , canonicalheading / font .value_required:n = true
92 , canonicalheading / font .initial:n = \footnotesize
93 , canonicalheading / format .cs_set:Np =
94   \__scripture_canonicalheading_format:n #1
95 , canonicalheading / format .value_required:n = true
96 , canonicalheading / format .initial:n = #1
97 , colour .tl_set:N = \l__scripture_colour_tl
98 , colour .value_required:n = true
99 , colour .initial:n = .
100 , color .meta:n = { colour = {#1} }
101 , compact .bool_set:N = \l__scripture_compact_bool
102 , compact .default:n = true
103 , compact .initial:n = false
104 , chapter .int_gset:N = \g__scripture_chapter_int
105 , chapter .value_required:n = true
106 , chapter .initial:n = \c_one_int
107 , chapter / aboveskip .tl_set:N = \l__scripture_chapter_aboveskip_tl
108 , chapter / aboveskip .value_required:n = true
109 , chapter / aboveskip .initial:n = \c_zero_skip
110 , chapter / align .choices:nn =
111   { left, right }

```

```

112     {
113         \tl_if_exist:NF \l__scripture_chapter_align_tl
114         { \tl_new:N \l__scripture_chapter_align_tl }
115         \str_case:Vn \l_keys_choice_str
116         {
117             { left }
118             {
119                 \tl_set:Nn \l__scripture_chapter_align_tl
120                 { l }
121             }
122             { right }
123             {
124                 \tl_set:Nn \l__scripture_chapter_align_tl
125                 { r }
126             }
127         }
128     }
129 , chapter / align .value_required:n = true
130 , chapter / align .initial:n = left
131 , chapter / font .tl_set:N = \l__scripture_chap_font_tl
132 , chapter / font .value_required:n = true
133 , chapter / font .initial:n = \bfseries
134 , chapter / colour .tl_set:N = \l__scripture_chapter_colour_tl
135 , chapter / colour .value_required:n = true
136 , chapter / colour .initial:n = .
137 , chapter / color .meta:n = { chapter / colour = {#1} }
138 , chapter / drop .bool_set:N = \l__scripture_chap_drop_bool
139 , chapter / drop .default:n = true
140 , chapter / drop .initial:n = true
141 , chapter / format .cs_set:Np = \__scripture_chap_format:n #1
142 , chapter / format .value_required:n = true
143 , chapter / format .initial:n = #1
144 , chapter / hide .bool_set_inverse:N = \l__scripture_chap_show_bool
145 , chapter / hide .default:n = true
146 , chapter / language .tl_set:N = \l__scripture_chapter_language_tl
147 , chapter / language .value_required:n = true
148 , chapter / language .initial:n =
149 , chapter / language / variant .tl_set:N =
150     \l__scripture_chapter_language_variant_tl
151 , chapter / language / variant .value_required:n = true
152 , chapter / language / variant .initial:n =
153 , chapter / nodrop .bool_set_inverse:N = \l__scripture_chap_drop_bool
154 , chapter / nodrop .default:n = true
155 , chapter / para .bool_set:N = \l__scripture_chap_para_bool
156 , chapter / para .default:n = true
157 , chapter / para .initial:n = false
158 , chapter / para / aboveskip .tl_set:N =
159     \l__scripture_chap_para_aboveskip_tl
160 , chapter / para / aboveskip .value_required:n = true
161 , chapter / para / aboveskip .initial:n = \bigskipamount
162 , chapter / para / belowskip .tl_set:N =
163     \l__scripture_chap_para_belowskip_tl
164 , chapter / para / belowskip .value_required:n = true
165 , chapter / para / belowskip .initial:n = \medskipamount

```

```

166 , chapter / para / indent .code:n =
167 {
168     \bool_set:Nn
169     \l__scripture_chap_para_afterindent_bool
170     { \str_if_eq_p:nn {#1} { true } }
171 }
172 , chapter / para / indent .default:n = true
173 , chapter / para / indent .initial:n = true
174 , chapter / presep .tl_set:N = \l__scripture_chap_presep_tl
175 , chapter / presep .value_required:n = true
176 , chapter / presep .initial:n = 0em
177 , chapter / sep .tl_set:N = \l__scripture_chap_sep_tl
178 , chapter / sep .value_required:n = true
179 , chapter / sep .initial:n = 0.5em
180 , chapter / show .bool_set:N = \l__scripture_chap_show_bool
181 , chapter / show .default:n = true
182 , chapter / show .initial:n = true
183 , chapter / showverse .bool_set:N = \l__scripture_chap_show_verse_bool
184 , chapter / showverse .default:n = true
185 , chapter / showverse .initial:n = false
186 , chapter / smash .bool_set:N = \l__scripture_chap_smash_bool
187 , chapter / smash .default:n = true
188 , chapter / smash .initial:n = false
189 , chapter / valign .choices:nn =
190 { bottom, middle, top }
191 {
192     \tl_if_exist:NF \l__scripture_chapter_valign_tl
193     { \tl_new:N \l__scripture_chapter_valign_tl }
194     \str_case:Vn \l_keys_choice_str
195     {
196         { bottom }
197         {
198             \tl_set:Nn \l__scripture_chapter_valign_tl
199             { b }
200         }
201         { middle }
202         {
203             \tl_set:Nn \l__scripture_chapter_valign_tl
204             { vc }
205         }
206         { top }
207         {
208             \tl_set:Nn \l__scripture_chapter_valign_tl
209             { t }
210         }
211     }
212 }
213 , chapter / valign .value_required:n = true
214 , chapter / valign .initial:n = bottom
215 , chapter / xchar .tl_set:N = \l__scripture_X_char_tl
216 , chapter / xchar .value_required:n = true
217 , chapter / xchar .initial:n = X
218 , defaults .code:n = \cs_if_exist_use:N \__scripture_setup_reset_defaults:
219 , defaults .value_forbidden:n = true

```

```

220 , extraskip .tl_set:N = \l__scripture_extraskip_tl
221 , extraskip .value_required:n = true
222 , extraskip .initial:n = \medskipamount
223 , font .tl_set:N = \l__scripture_font_tl
224 , font .value_required:n = true
225 , font .initial:n =
226 , heading / aboveskip .tl_set:N = \l__scripture_heading_aboveskip_tl
227 , heading / aboveskip .value_required:n = true
228 , heading / aboveskip .initial:n = \bigskipamount
229 , heading / afterindent .bool_set:N = \l__scripture_heading_afterindent_bool
230 , heading / afterindent .default:n = true
231 , heading / afterindent .initial:n = false
232 , heading / align .choices:nn =
233 { left, right, center }
234 {
235   \tl_if_exist:NF \l__scripture_heading_align_tl
236   { \tl_new:N \l__scripture_heading_align_tl }
237   \str_case:Vn \l_keys_choice_str
238   {
239     { left }
240     {
241       \tl_set:Nn \l__scripture_heading_align_tl
242       { \raggedright }
243     }
244     { right }
245     {
246       \tl_set:Nn \l__scripture_heading_align_tl
247       { \raggedleft }
248     }
249     { center }
250     {
251       \tl_set:Nn \l__scripture_heading_align_tl
252       { \centering }
253     }
254   }
255 }
256 , heading / align .value_required:n = true
257 , heading / align .initial:n = left
258 , heading / belowskip .tl_set:N = \l__scripture_heading_belowskip_tl
259 , heading / belowskip .value_required:n = true
260 , heading / belowskip .initial:n = \medskipamount
261 , heading / font .tl_set:N = \l__scripture_heading_font_tl
262 , heading / font .value_required:n = true
263 , heading / font .initial:n = \small\itshape
264 , heading / format .cs_set:Np = \__scripture_heading_format:n #1
265 , heading / format .value_required:n = true
266 , heading / format .initial:n = #1
267 , heading / hide .bool_set_inverse:N = \l__scripture_heading_show_bool
268 , heading / hide .default:n = true
269 , heading / show .bool_set:N = \l__scripture_heading_show_bool
270 , heading / show .default:n = true
271 , heading / show .initial:n = true
272 , indent .bool_set:N = \l__scripture_indent_bool
273 , indent .default:n = true

```

```

274 , indent .initial:n = true
275 , inline .bool_set:N = \l__scripture_inline_bool
276 , inline .default:n = true
277 , inline .initial:n = false
278 , inline / begin .tl_set:N = \l__scripture_inline_begin_tl
279 , inline / begin .value_required:n = true
280 , inline / begin .initial:n = ``
281 , inline / end .tl_set:N = \l__scripture_inline_end_tl
282 , inline / end .value_required:n = true
283 , inline / end .initial:n = ''
284 , inline / font .tl_set:N = \l__scripture_inline_font_tl
285 , inline / font .value_required:n = true
286 , inline / font .initial:n =
287 , inline / reference / format .cs_set:Np =
288   \__scripture_inline_ref_format:n #1
289 , inline / reference / format .value_required:n = true
290 , inline / reference / format .initial:n = (#1)
291 , inline / reference / sep .tl_set:N = \l__scripture_inline_ref_sep_tl
292 , inline / reference / sep .value_required:n = true
293 , inline / reference / sep .initial:n = 0.5em
294 , inline / version / delim .tl_set:N = \l__scripture_inline_version_delim_tl
295 , inline / version / delim .value_required:n = true
296 , inline / version / delim .initial:n = \c_space_tl
297 , inline / version / format .cs_set:Np =
298   \__scripture_inline_version_format:n #1
299 , inline / version / format .value_required:n = true
300 , inline / version / format .initial:n = #1
301 , language .tl_set:N = \l__scripture_main_language_tl
302 , language .value_required:n = true
303 , language .initial:n =
304 , language / variant .tl_set:N = \l__scripture_main_language_variant_tl
305 , language / variant .value_required:n = true
306 , language / variant .initial:n =
307 , leftmargin .tl_set:N = \l__scripture_leftmargin_tl
308 , leftmargin .value_required:n = true
309 , leftmargin .initial:n = \c_zero_dim
310 , mark / verse / format .cs_set:Np = \__scripture_mark_verse_format:nnn
311   #1#2#3
312 , mark / verse / format .value_required:n = true
313 , mark / verse / format .initial:n = #1 ~ #2:#3
314 , name / font .tl_set:N = \l__scripture_name_font_tl
315 , name / font .value_required:n = true
316 , name / font .initial:n = \scshape
317 , name / format .cs_set:Np = \__scripture_name_format:n #1
318 , name / format .value_required:n = true
319 , name / format .initial:n = #1
320 , noindent .bool_set_inverse:N = \l__scripture_indent_bool
321 , noindent .default:n = true
322 , parindent .tl_set:N = \l__scripture_parindent_tl
323 , parindent .value_required:n = true
324 , parindent .initial:n = \parindent
325 , parskip .tl_set:N = \l__scripture_parskip_tl
326 , parskip .value_required:n = true
327 , parskip .initial:n = \parskip

```



```

328 , pilcrow .bool_set:N = \l__scripture_pilcrow_bool
329 , pilcrow .default:n = true
330 , pilcrow .initial:n = false
331 , pilcrow / sep .tl_set:N = \l__scripture_pilcrow_sep_tl
332 , pilcrow / sep .value_required:n = true
333 , pilcrow / sep .initial:n = 0.25em
334 , redletter .bool_set:N = \l__scripture_red_letter_bool
335 , redletter .default:n = true
336 , redletter .initial:n = false
337 , redletter / colour .tl_set:N = \l__scripture_red_letter_colour_tl
338 , redletter / colour .value_required:n = true
339 , redletter / colour .initial:n = red!80!black
340 , redletter / color .meta:n = { redletter / colour = {#1} }
341 , reference / align .choices:nn =
342   { left, right }
343   {
344     \str_if_exist:NF \l__scripture_ref_align_str
345       { \str_new:N \l__scripture_ref_align_str }
346     \str_set_eq:NN \l__scripture_ref_align_str \l_keys_choice_str
347   }
348 , reference / align .value_required:n = true
349 , reference / align .initial:n = right
350 , reference / colour .tl_set:N = \l__scripture_ref_colour_tl
351 , reference / colour .value_required:n = true
352 , reference / colour .initial:n = .
353 , reference / color .meta:n = { reference / colour = {#1} }
354 , reference / font .tl_set:N = \l__scripture_ref_font_tl
355 , reference / font .value_required:n = true
356 , reference / font .initial:n = \bfseries
357 , reference / format .cs_set:Np = \__scripture_ref_format:n #1
358 , reference / format .value_required:n = true
359 , reference / format .initial:n = #1
360 , reference / language .tl_set:N = \l__scripture_reference_language_tl
361 , reference / language .value_required:n = true
362 , reference / language .initial:n =
363 , reference / language / variant .tl_set:N =
364   \l__scripture_reference_language_variant_tl
365 , reference / language / variant .value_required:n = true
366 , reference / language / variant .initial:n =
367 , reference / newline .meta:n = { reference / sep = \linewidth }
368 , reference / newline .value_forbidden:n = true
369 , reference / position .choices:nn =
370   { start, end }
371   {
372     \str_if_exist:NF \l__scripture_ref_position_str
373       { \str_new:N \l__scripture_ref_position_str }
374     \str_set_eq:NN \l__scripture_ref_position_str \l_keys_choice_str
375   }
376 , reference / position .value_required:n = true
377 , reference / position .initial:n = end
378 , reference / sep .tl_set:N = \l__scripture_ref_sep_tl
379 , reference / sep .value_required:n = true
380 , reference / sep .initial:n = 2em
381 , reference / start / inline .bool_set_inverse:N =

```

```

382     \l__scripture_ref_start_newline_bool
383 , reference / start / inline .default:n = true
384 , reference / start / newline .bool_set:N =
385     \l__scripture_ref_start_newline_bool
386 , reference / start / newline .default:n = true
387 , reference / start / newline .initial:n = false
388 , reference / start / sep .tl_set:N = \l__scripture_ref_start_sep_tl
389 , reference / start / sep .value_required:n = true
390 , reference / start / sep .initial:n = 1em
391 , reference / start / vsep .tl_set:N = \l__scripture_ref_vertical_sep_tl
392 , reference / start / vsep .value_required:n = true
393 , reference / start / vsep .initial:n = Opt
394 , rightmargin .tl_set:N = \l__scripture_rightmargin_tl
395 , rightmargin .value_required:n = true
396 , rightmargin .initial:n = \c_zero_dim
397 , selah / text .tl_set:N = \l__scripture_selah_text_tl
398 , selah / text .value_required:n = true
399 , selah / text .initial:n = Selah
400 , selah / font .tl_set:N = \l__scripture_selah_font_tl
401 , selah / font .value_required:n = true
402 , selah / font .initial:n = \itshape
403 , selah / format .cs_set:Np = \__scripture_selah_format:n #1
404 , selah / format .value_required:n = true
405 , selah / format .initial:n = #1
406 , selah / sep .tl_set:N = \l__scripture_selah_sep_tl
407 , selah / sep .value_required:n = true
408 , selah / sep .initial:n = 1em
409 , style .choice:
410 , style / unknown .code:n = \msg_error:nnx
411     { scripture }
412     { unknown-style }
413     { \exp_not:n {#1} }
414 , textright / sep .tl_set:N = \l__scripture_text_right_sep_tl
415 , textright / sep .value_required:n = true
416 , textright / sep .initial:n = 1em
417 , verse .int_gset:N = \g__scripture_verse_int
418 , verse .value_required:n = true
419 , verse .initial:n = \c_one_int
420 , verse / colour .tl_set:N = \l__scripture_verse_colour_tl
421 , verse / colour .value_required:n = true
422 , verse / colour .initial:n = .
423 , verse / color .meta:n = { verse / colour = {#1} }
424 , verse / first .bool_set:N = \l__scripture_verse_first_bool
425 , verse / first .default:n = true
426 , verse / first .initial:n = false
427 , verse / firstformat .cs_set:Np = \__scripture_verse_first_format:n #1
428 , verse / firstformat .value_required:n = true
429 , verse / firstformat .initial:n = #1
430 , verse / firstsep .tl_set:N = \l__scripture_verse_first_sep_tl
431 , verse / firstsep .value_required:n = true
432 , verse / firstsep .initial:n = 0.5em
433 , verse / font .tl_set:N = \l__scripture_verse_font_tl
434 , verse / font .value_required:n = true
435 , verse / font .initial:n =

```

```

436 , verse / format .cs_set:Np = \__scripture_verse_format:n #1
437 , verse / format .value_required:n = true
438 , verse / format .initial:n = \textsuperscript{#1}
439 , verse / hide .bool_set_inverse:N = \l__scripture_verse_show_bool
440 , verse / hide .default:n = true
441 , verse / language .tl_set:N = \l__scripture_verse_language_tl
442 , verse / language .value_required:n = true
443 , verse / language .initial:n =
444 , verse / language / variant .tl_set:N =
445     \l__scripture_verse_language_variant_tl
446 , verse / language / variant .value_required:n = true
447 , verse / language / variant .initial:n =
448 , verse / para .bool_set:N = \l__scripture_verse_para_bool
449 , verse / para .default:n = true
450 , verse / para .initial:n = false
451 , verse / para / indent .tl_set:N = \l__scripture_para_indent_tl
452 , verse / para / indent .value_required:n = true
453 , verse / para / indent .initial:n = \l__scripture_parindent_tl
454 , verse / presep .tl_set:N = \l__scripture_verse_presep_tl
455 , verse / presep .value_required:n = true
456 , verse / presep .initial:n = 0em
457 , verse / sep .tl_set:N = \l__scripture_verse_sep_tl
458 , verse / sep .value_required:n = true
459 , verse / sep .initial:n = 0.05em
460 , verse / show .bool_set:N = \l__scripture_verse_show_bool
461 , verse / show .default:n = true
462 , verse / show .initial:n = true
463 , version .tl_set:N = \l__scripture_version_tl
464 , version .value_required:n = true
465 , version .initial:n =
466 , version / delim .tl_set:N = \l__scripture_version_delim_tl
467 , version / delim .value_required:n = true
468 , version / delim .initial:n = \c_space_tl
469 , version / format .cs_set:Np = \__scripture_version_format:n #1
470 , version / format .value_required:n = true
471 , version / format .initial:n = (#1)
472 , version / language .tl_set:N = \l__scripture_version_language_tl
473 , version / language .value_required:n = true
474 , version / language .initial:n =
475 , version / language / variant .tl_set:N =
476     \l__scripture_version_language_variant_tl
477 , version / language / variant .value_required:n = true
478 , version / language / variant .initial:n =
479 , version / position .choices:nn =
480     { withref, end }
481     {
482         \str_if_exist:NF \l__scripture_version_position_str
483         { \str_new:N \l__scripture_version_position_str }
484         \str_set_eq:NN \l__scripture_version_position_str \l_keys_choice_str
485     }
486 , version / position .value_required:n = true
487 , version / position .initial:n = withref

```

Options for the `center` environment.

```

488 , center / aboveskip .tl_set:N = \l__scripture_center_aboveskip_tl
489 , center / aboveskip .value_required:n = true
490 , center / aboveskip .initial:n = \medskipamount
491 , center / belowskip .tl_set:N = \l__scripture_center_belowskip_tl
492 , center / belowskip .value_required:n = true
493 , center / belowskip .initial:n = \medskipamount
494 , center / leftmargin .tl_set:N = \l__scripture_center_leftmargin_tl
495 , center / leftmargin .value_required:n = true
496 , center / leftmargin .initial:n = \c_zero_dim
497 , center / rightmargin .tl_set:N = \l__scripture_center_rightmargin_tl
498 , center / rightmargin .value_required:n = true
499 , center / rightmargin .initial:n = \c_zero_dim

```

Options for the **flushright** environment.

```

500 , flushright / aboveskip .tl_set:N = \l__scripture_flushright_aboveskip_tl
501 , flushright / aboveskip .value_required:n = true
502 , flushright / aboveskip .initial:n = \medskipamount
503 , flushright / belowskip .tl_set:N = \l__scripture_flushright_belowskip_tl
504 , flushright / belowskip .value_required:n = true
505 , flushright / belowskip .initial:n = \medskipamount
506 , flushright / leftmargin .tl_set:N = \l__scripture_flushright_leftmargin_tl
507 , flushright / leftmargin .value_required:n = true
508 , flushright / leftmargin .initial:n = \c_zero_dim
509 , flushright / rightmargin .tl_set:N =
510   \l__scripture_flushright_rightmargin_tl
511 , flushright / rightmargin .value_required:n = true
512 , flushright / rightmargin .initial:n = \c_zero_dim

```

Options for the **hanging** environment.

```

513 , hanging / aboveskip .tl_set:N = \l__scripture_hanging_aboveskip_tl
514 , hanging / aboveskip .value_required:n = true
515 , hanging / aboveskip .initial:n = \medskipamount
516 , hanging / belowskip .tl_set:N = \l__scripture_hanging_belowskip_tl
517 , hanging / belowskip .value_required:n = true
518 , hanging / belowskip .initial:n = \medskipamount
519 , hanging / chapter / indent .bool_set:N =
520   \l__scripture_hanging_chapter_indent_bool
521 , hanging / chapter / indent .default:n = true
522 , hanging / chapter / indent .initial:n = true
523 , hanging / hang .tl_set:N = \l__scripture_hanging_hang_tl
524 , hanging / hang .value_required:n = true
525 , hanging / hang .initial:n = 2em
526 , hanging / leftmargin .tl_set:N = \l__scripture_hanging_leftmargin_tl
527 , hanging / leftmargin .value_required:n = true
528 , hanging / leftmargin .initial:n = 1em
529 , hanging / rightmargin .tl_set:N = \l__scripture_hanging_rightmargin_tl
530 , hanging / rightmargin .value_required:n = true
531 , hanging / rightmargin .initial:n = \c_zero_dim
532 , hanging / verse / left .bool_set_inverse:N =
533   \l__scripture_hanging_verse_right_bool
534 , hanging / verse / left .default:n = true
535 , hanging / verse / right .bool_set:N =
536   \l__scripture_hanging_verse_right_bool
537 , hanging / verse / right .default:n = true
538 , hanging / verse / right .initial:n = true

```

```

539 , hanging / verse / sep .tl_set:N = \l__scripture_hanging_verse_sep_tl
540 , hanging / verse / sep .value_required:n = true
541 , hanging / verse / sep .initial:n = 0.05em

```

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

```

542 , midparachap / hideverse .bool_set_inverse:N =
543     \l__scripture_midparachap_show_verse_bool
544 , midparachap / hideverse .default:n = true
545 , midparachap / showverse .bool_set:N =
546     \l__scripture_midparachap_show_verse_bool
547 , midparachap / showverse .default:n = true
548 , midparachap / showverse .initial:n = true

```

Options for the `narrow` environment.

```

549 , narrow / aboveskip .tl_set:N = \l__scripture_narrow_aboveskip_tl
550 , narrow / aboveskip .value_required:n = true
551 , narrow / aboveskip .initial:n = \medskipamount
552 , narrow / belowskip .tl_set:N = \l__scripture_narrow_belowskip_tl
553 , narrow / belowskip .value_required:n = true
554 , narrow / belowskip .initial:n = \medskipamount
555 , narrow / leftmargin .tl_set:N = \l__scripture_narrow_leftmargin_tl
556 , narrow / leftmargin .value_required:n = true
557 , narrow / leftmargin .initial:n = 1em
558 , narrow / rightmargin .tl_set:N = \l__scripture_narrow_rightmargin_tl
559 , narrow / rightmargin .value_required:n = true
560 , narrow / rightmargin .initial:n = 1em

```

Options for the `poetry` environment.

```

561 , poetry / aboveskip .tl_set:N = \l__scripture_poetry_aboveskip_tl
562 , poetry / aboveskip .value_required:n = true
563 , poetry / aboveskip .initial:n = \medskipamount
564 , poetry / belowskip .tl_set:N = \l__scripture_poetry_belowskip_tl
565 , poetry / belowskip .value_required:n = true
566 , poetry / belowskip .initial:n = \medskipamount
567 , poetry / bigindent .tl_set:N = \l__scripture_poetry_big_indent_tl
568 , poetry / bigindent .value_required:n = true
569 , poetry / bigindent .initial:n = 4em
570 , poetry / indent .tl_set:N = \l__scripture_poetry_indent_tl
571 , poetry / indent .value_required:n = true
572 , poetry / indent .initial:n = 1em
573 , poetry / leftmargin .tl_set:N = \l__scripture_poetry_leftmargin_tl
574 , poetry / leftmargin .value_required:n = true
575 , poetry / leftmargin .initial:n = 1em
576 , poetry / rightmargin .tl_set:N = \l__scripture_poetry_rightmargin_tl
577 , poetry / rightmargin .value_required:n = true
578 , poetry / rightmargin .initial:n = \c_zero_dim
579 , poetry / secondhalf .bool_set:N = \l__scripture_poetry_secondhalf_bool
580 , poetry / secondhalf .default:n = true
581 , poetry / secondhalf .initial:n = false
582 , poetry / verse / left .bool_set_inverse:N =
583     \l__scripture_poetry_verse_right_bool
584 , poetry / verse / left .default:n = true
585 , poetry / verse / right .bool_set:N = \l__scripture_poetry_verse_right_bool
586 , poetry / verse / right .default:n = true
587 , poetry / verse / right .initial:n = true

```

```

588     , poetry / verse / sep .tl_set:N = \l__scripture_poetry_verse_sep_tl
589     , poetry / verse / sep .value_required:n = true
590     , poetry / verse / sep .initial:n = 0.05em
591 }

```

Process package options.

```

592 \ProcessKeyOptions
    Create interface to reset all keys to defaults.
593 \tl_new:N \l__scripture_default_options_tl
594 \keys_precompile:nnN { scripture }
595 {
596     , aboveskip                = \c_zero_skip
597     , added / font             =
598     , added / format           = \emph {#1}
599     , after                    =
600     , before                   =
601     , belowskip                = \c_zero_skip
602     , book                    =
603     , canonicalheading / aboveskip = \medskipamount
604     , canonicalheading / afterindent = false
605     , canonicalheading / align  = center
606     , canonicalheading / belowskip = \medskipamount
607     , canonicalheading / font    = \footnotesize
608     , colour                   = .
609     , compact                  = false
610     , chapter                  = \c_one_int
611     , chapter / aboveskip      = \c_zero_skip
612     , chapter / align          = left
613     , chapter / font           = \bfseries
614     , chapter / colour         = .
615     , chapter / drop           = true
616     , chapter / format         = #1
617     , chapter / para           = false
618     , chapter / para / aboveskip = \bigskipamount
619     , chapter / para / belowskip = \medskipamount
620     , chapter / para / indent  = true
621     , chapter / presep         = 0em
622     , chapter / sep            = 0.5em
623     , chapter / show           = true
624     , chapter / showverse      = false
625     , chapter / smash          = false
626     , chapter / valign         = bottom
627     , chapter / xchar          = X
628     , extraskip                = \medskipamount
629     , font                     =
630     , heading / aboveskip      = \bigskipamount
631     , heading / afterindent    = false
632     , heading / align          = left
633     , heading / belowskip      = \medskipamount
634     , heading / font           = \small\itshape
635     , heading / format         = #1
636     , heading / show           = true
637     , indent                   = true
638     , inline                   = false

```

```

639 , inline / begin = ``
640 , inline / end = ''
641 , inline / font =
642 , inline / reference / format = (#1)
643 , inline / reference / sep = 0.5em
644 , inline / version / delim = \c_space_tl
645 , inline / version / format = #1
646 , language =
647 , language / variant =
648 , leftmargin = \c_zero_dim
649 , mark / verse / format = #1 #2:#3
650 , name / font = \scshape
651 , name / format = #1
652 , parindent = \parindent
653 , parskip = \parskip
654 , pilcrow = false
655 , pilcrow / sep = 0.25em
656 , redletter = false
657 , redletter / colour = red!80!black
658 , reference / align = right
659 , reference / colour = .
660 , reference / font = \bfseries
661 , reference / format = #1
662 , reference / language =
663 , reference / language / variant =
664 , reference / position = end
665 , reference / sep = 2em
666 , reference / start / newline = false
667 , reference / start / sep = 1em
668 , reference / start / vsep = 0pt
669 , rightmargin = \c_zero_dim
670 , selah / text = Selah
671 , selah / font = \itshape
672 , selah / format = #1
673 , selah / sep = 1em
674 , textright / sep = 1em
675 , verse = \c_one_int
676 , verse / colour = .
677 , verse / first = false
678 , verse / firstformat = #1
679 , verse / firstsep = 0.5em
680 , verse / font =
681 , verse / format = \textsuperscript{#1}
682 , verse / language =
683 , verse / language / variant =
684 , verse / para = false
685 , verse / para / indent = \l__scripture_parindent_tl
686 , verse / presep = 0em
687 , verse / sep = 0.05em
688 , verse / show = true
689 , version =
690 , version / delim = \c_space_tl
691 , version / format = (#1)
692 , version / language =

```

```

693 , version / language / variant =
694 , version / position = withref
695 , center / aboveskip = \medskipamount
696 , center / belowskip = \medskipamount
697 , center / leftmargin = \c_zero_dim
698 , center / rightmargin = \c_zero_dim
699 , flushright / aboveskip = \medskipamount
700 , flushright / belowskip = \medskipamount
701 , flushright / leftmargin = \c_zero_dim
702 , flushright / rightmargin = \c_zero_dim
703 , hanging / aboveskip = \medskipamount
704 , hanging / belowskip = \medskipamount
705 , hanging / chapter / indent = true
706 , hanging / hang = 2em
707 , hanging / leftmargin = 1em
708 , hanging / rightmargin = \c_zero_dim
709 , hanging / verse / right = true
710 , hanging / verse / sep = 0.05em
711 , midparachap / showverse = true
712 , narrow / aboveskip = \medskipamount
713 , narrow / belowskip = \medskipamount
714 , narrow / leftmargin = 1em
715 , narrow / rightmargin = 1em
716 , poetry / aboveskip = \medskipamount
717 , poetry / belowskip = \medskipamount
718 , poetry / bigindent = 4em
719 , poetry / indent = 1em
720 , poetry / leftmargin = 1em
721 , poetry / rightmargin = \c_zero_dim
722 , poetry / secondhalf = false
723 , poetry / verse / right = true
724 , poetry / verse / sep = 0.05em
725 }
726 \l__scripture_default_options_tl
727 \cs_new_protected:Nn \__scripture_setup_reset_defaults:
728 {
729   \l__scripture_default_options_tl
730 }

```

`\__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.

```

731 \cs_new_protected:Nn \__scripture_setup_style:nnn
732 {
733   \bool_if:NTF #1
734   {
735     \tl_if_exist:cF { g__scripture_style_ #2 _tl }
736     { \tl_new:c { g__scripture_style_ #2 _tl } }
737     \__scripture_setup_style:nne { \c_false_bool } {#2}
738     { \tl_use:c { g__scripture_style_ #2 _tl } , #3 }
739   }
740   {
741     \tl_gset:cn { g__scripture_style_ #2 _tl } {#3}
742     \keys_define:nn { scripture }
743     { style / #2 .meta:n = {#3} }

```



```

744     }
745   }
746   \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.

```

747 \hook_new_pair:nn { scripture / chap / before } { scripture / chap / after }
748 \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. Pilecrows are activated using the **pilcrow** option and are only in effect when the **verse/para** and/or **compact** options are in use.

```

749 \hook_new:n { scripture / pilcrow }
750 \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.

```

751 \mark_new_class:n { scripture / heading }
752 \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:

```

753 \cs_new_protected:Nn \__scripture_insert_verse_mark:
754 {
755   \tl_if_empty:NF \scripturecurrentbook
756   {
757     \mark_insert:nn { scripture / verse }
758     {
759       \__scripture_mark_verse_format:nnn
760       { \scripturecurrentbook }
761       { \scripturecurrentchapter }
762       { \scripturecurrentverse }
763     }
764   }
765 }

```

(End of definition for `\_scripture_insert_verse_mark:.`)

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

```

766 \hook_gput_code:nnn { insertmark } { scripture }
767 {
768   \cs_set_eq:NN \name \_scripture_name:n
769   \cs_set_nopar:Npn \LORD { \_scripture_name:n { Lord } }
770   \cs_set_nopar:Npn \GOD { \_scripture_name:n { God } }
771 }

```

3.5 The main **scripture** environment

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

```

\l__scripture_active_inner_bool
772 \bool_new:N \l__scripture_active_bool
773 \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.

```

774 \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.

```

775 \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.

```

776 \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.

```

777 \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.

```

778 \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.

```

779 \cs_new_protected_nopar:Nn \_scripture_setup_list_noindent:n
780 {
781   \legacy_if:nT { @newlist }
782   {
783     \cs_set_protected_nopar:Npn \noindent

```

```

784 {
785   \skip_horizontal:n { -#1 }
786   \bool_if:NT \l__scripture_verse_para_bool
787   {
788     \peek_analysis_map_inline:n
789     {
790       \bool_lazy_or:nnTF
791       { \exp_args:No \token_if_eq_meaning_p:NN {####1} \vs }
792       { \exp_args:No \token_if_eq_meaning_p:NN {####1} \newvs }
793       {
794         \bool_gset_true:N
795         \g__scripture_suppress_next_verse_para_bool
796         \__scripture_set_pilcrow_hook:
797       }
798       {
799         \str_if_eq:VnF \l__scripture_currenvir_str { poetry }
800         { \__scripture_pilcrow_output: }
801       }
802       \peek_analysis_map_break:n {####1}
803     }
804   }
805 }
806 \hook_gput_next_code:nn { para / end }
807 {
808   \bool_if:NTF \l__scripture_verse_para_bool
809   {
810     \str_if_eq:VnTF \l__scripture_currenvir_str { poetry }
811     { \cs_set_eq:NN \noindent \__scripture_noindent_saved: }
812     { \cs_set_eq:NN \noindent \__scripture_para_noindent: }
813   }
814   { \cs_set_eq:NN \noindent \__scripture_noindent_saved: }
815 }
816 }
817 }

```

(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.

```

818 \cs_new_protected:Nn \__scripture_skip_diff:NN
819 {
820   \skip_eval:n
821   {
822     \dim_eval:n { \dimexpr #1 - \dimexpr #2 }
823     ~ plus ~
824     \dim_max:nn { \gluestretch #1 - \gluestretch #2 } \c_zero_dim
825     ~ minus ~
826     \dim_max:nn { \glueshrink #1 - \glueshrink #2 } \c_zero_dim
827   }
828 }

```

(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.

```

829 \cs_new_protected:Nn \__scripture_extra_skip:
830 {
831   \mode_if_vertical:F
832   {
833     \bool_if:NTF \l__scripture_verse_para_bool
834     { \para_end: }
835     { \par }
836   }
837   \int_compare:nNnT
838   { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
839   {
840     \dim_compare:nNnT
841     { \parskip + \glueexpr \l__scripture_extraskip_tl }
842     <
843     { \baselineskip }
844     {
845       \addvspace { \baselineskip }
846     }
847     \int_gzero:N \g__scripture_chap_par_prevgraf_int
848   }
849   \addvspace { \l__scripture_extraskip_tl }
850 }

```

(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.

```

851 \cs_new_protected:Nn \__scripture_text_right:nn
852 {
853   \bool_lazy_or:nnTF
854   { \str_if_eq_p:Vn \l__scripture_currenvir_str { flushright } }
855   { \str_if_eq_p:Vn \l__scripture_currenvir_str { center } }
856   {
857     \unskip
858     \break
859     \strut
860     \hfill
861     #2
862     \para_end:
863   }
864   {
865     {
866       \unskip
867       \hfil
868       \penalty 50
869       \skip_horizontal:n {#1}
870       \hbox:n {}
871       \nobreak
872       \hfill
873       #2
874       \skip_zero:N \parfillskip
875       \int_zero:N \finalhyphendemerits

```

```

876         \para_end:
877     }
878 }
879 }

```

(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.

```

880 \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.

```

881 \cs_new_protected:Nn \_scripture_format_full_ref:nn
882 {
883     \group_begin:
884     \l\_scripture_ref_font_tl
885     \str_if_eq:VnF \l\_scripture_ref_colour_tl { . }
886     { \exp_args:NV \color_select:n \l\_scripture_ref_colour_tl }
887     \_scripture_select_language:n { reference }
888     \_scripture_ref_format:n
889     {
890         #1

```

Append a Bible version if it has been specified.

```

891         \tl_if_empty:oF {#2}
892         {
893             \tl_if_empty:oF {#1}
894             { \l\_scripture_version_delim_tl }
895             \_scripture_select_language:n { version }
896             \_scripture_version_format:n {#2}
897         }
898     }
899     \group_end:
900 }

```

(End of definition for _scripture_format_full_ref:nn.)

_scripture_reference:n Output the reference for the Scripture quotation.

```

901 \cs_new_protected:Nn \_scripture_reference:n
902 {
903     \hbox_set:Nn \l\_scripture_ref_box
904     {
905         \_scripture_format_full_ref:nn {#1} \l\_scripture_version_tl
906     }
907     \dim_compare:nNnT
908     {
909         \box_wd:N \l\_scripture_ref_box +
910         \g\_scripture_final_line_dim + \l\_scripture_ref_sep_tl
911     }
912     >
913     { \linewidth }
914     {

```

```

915     \skip_vertical:N \baselineskip
916     \str_if_eq:VnT \l__scripture_ref_align_str { left }
917     {
918         \int_compare:nNnT
919         { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
920         { \skip_vertical:N \baselineskip }
921     }
922 }
923 \str_if_eq:VnTF \l__scripture_ref_align_str { right }
924 { \hfill }
925 {
926     \dim_compare:nNnT
927     {
928         \box_wd:N \l__scripture_ref_box +
929         \g__scripture_final_line_dim + \l__scripture_ref_sep_tl
930     }
931     <
932     { \linewidth }
933     {
934         \skip_horizontal:n
935         { \g__scripture_final_line_dim + \l__scripture_ref_sep_tl }
936     }
937 }
938 \box_use:N \l__scripture_ref_box
939 }

```

(End of definition for __scripture_reference:n.)

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

```

940 \cs_new_protected:Nn \__scripture_reference_start_inline:
941 {
942     \tl_if_empty:NF \l__scripture_ref_tl
943     {
944         \str_if_eq:VnTF \l__scripture_version_position_str { withref }
945         {
946             \__scripture_format_full_ref:nn
947             { \l__scripture_ref_tl } { \l__scripture_version_tl }
948         }
949         { \__scripture_format_full_ref:nn { \l__scripture_ref_tl } { } }
950         \skip_horizontal:N \l__scripture_ref_start_sep_tl
951     }
952 }

```

(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.

```

953 \cs_new_protected:Nn \__scripture_reference_start_newline:
954 {
955     \tl_if_empty:NF \l__scripture_ref_tl
956     {
957         \bool_gset_eq:NN \g_tmpa_bool \g__scripture_suppress_next_pilcrow_bool
958         \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
959         \noindent
960         \bool_gset_eq:NN \g__scripture_suppress_next_pilcrow_bool \g_tmpa_bool
961         \str_if_eq:VnF \l__scripture_currenvir_str { scripture }

```

```

962     { \skip_horizontal:n { - \leftmargin } }
963 \str_if_eq:VnTF \l__scripture_version_position_str { withref }
964 {
965     \__scripture_format_full_ref:nn
966     { \l__scripture_ref_tl } { \l__scripture_version_tl }
967 }
968 { \__scripture_format_full_ref:nn { \l__scripture_ref_tl } { } }
969 \bool_lazy_or:nnT
970 { \str_if_eq_p:Vn \l__scripture_currenvir_str { flushright } }
971 { \str_if_eq_p:Vn \l__scripture_currenvir_str { center } }
972 {
973     \hfill
974     \strut
975 }
976 \bool_if:NTF \l__scripture_compact_bool
977 { \para_end: }
978 {
979     \bool_lazy_all:nTF
980     {
981         { \l__scripture_verse_para_bool }
982         {
983             \bool_not_p:n
984             { \str_if_eq_p:Vn \l__scripture_currenvir_str { hanging } }
985         }
986         {
987             \bool_not_p:n
988             { \str_if_eq_p:Vn \l__scripture_currenvir_str { poetry } }
989         }
990     }
991     { \para_end: }
992     { \par }
993 }
994 \nobreak
995 \@afterheading
996 \addvspace { \l__scripture_ref_vertical_sep_tl }
997 }
998 }

```

(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.

```

999 \cs_new_protected:Nn \__scripture_reference_start_peek:
1000 {
1001     \peek_remove_spaces:n
1002     {
1003         \bool_lazy_and:nnTF
1004         { \str_if_eq_p:Vn \l__scripture_ref_position_str { start } }
1005         { \bool_not_p:n { \tl_if_empty_p:N \l__scripture_ref_tl } }
1006         {
1007             \bool_lazy_or:nnF
1008             { \str_if_eq_p:Vn \l__scripture_currenvir_str { scripture } }
1009             { \str_if_eq_p:Vn \l__scripture_currenvir_str { midparachap } }
1010             {

```

```

1011         \bool_if:NF \l__scripture_compact_bool
1012         { \bool_set_true:N \l__scripture_ref_start_newline_bool }
1013     }
1014 \peek_analysis_map_inline:n
1015 {
1016     \bool_case:nF
1017     {
1018         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \heading }
1019         { \__scripture_reference_start_newline: }
1020         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \noindent }
1021         {
1022             \bool_if:NTF \l__scripture_ref_start_newline_bool
1023             { \__scripture_reference_start_newline: }
1024             { \__scripture_reference_start_inline: }
1025         }
1026         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nohang }
1027         {
1028             \bool_if:NTF \l__scripture_ref_start_newline_bool
1029             { \__scripture_reference_start_newline: }
1030             { \__scripture_reference_start_inline: }
1031         }
1032         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nopilcrow }
1033         {
1034             \bool_if:NTF \l__scripture_ref_start_newline_bool
1035             { \__scripture_reference_start_newline: }
1036             { \__scripture_reference_start_inline: }
1037             \peek_analysis_map_break:n
1038             {
1039                 ##1
1040                 \__scripture_fix_protrusion:n { }
1041             }
1042         }
1043         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \begin }
1044         {
1045             \bool_if:NF \l__scripture_indent_bool
1046             {
1047                 \hook_gput_next_code:nn { para / begin }
1048                 {
1049                     \bool_lazy_or:nnT
1050                     {
1051                         \str_if_eq_p:Vn
1052                         \l__scripture_currenvir_str
1053                         { midparachap }
1054                     }
1055                     {
1056                         \str_if_eq_p:Vn
1057                         \l__scripture_currenvir_str
1058                         { scripture }
1059                     }
1060                     { \para_omit_indent: }
1061                 }
1062             }
1063         }
1064         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \ch }

```



```

1065 {
1066     \bool_if:NTF \l__scripture_ref_start_newline_bool
1067     { \__scripture_reference_start_newline: }
1068     {
1069         \cs_set_eq:NN \__scripture_temp_ch \ch
1070         \RenewDocumentCommand \ch { s o m }
1071         {
1072             \tl_if_novalue:nF {##2}
1073             {
1074                 \tl_set:Nn \l__scripture_X_char_tl {##2}
1075             }
1076             \cs_set_eq:NN \ch \__scripture_temp_ch:
1077         }
1078         \__scripture_reference_start_inline:
1079     }
1080 }
1081 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newch }
1082 {
1083     \bool_if:NTF \l__scripture_ref_start_newline_bool
1084     { \__scripture_reference_start_newline: }
1085     {
1086         \cs_set_eq:NN \__scripture_temp_newch: \newch
1087         \RenewDocumentCommand \newch { s o }
1088         {
1089             \tl_if_novalue:nF {##2}
1090             {
1091                 \tl_set:Nn \l__scripture_X_char_tl {##2}
1092             }
1093             \cs_set_eq:NN \newch \__scripture_temp_newch:
1094         }
1095         \__scripture_reference_start_inline:
1096     }
1097 }
1098 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1099 {
1100     \bool_if:NTF \l__scripture_ref_start_newline_bool
1101     { \__scripture_reference_start_newline: }
1102     {
1103         \cs_set_eq:NN \__scripture_temp_vs: \vs
1104         \RenewDocumentCommand \vs { m }
1105         { \cs_set_eq:NN \vs \__scripture_temp_vs: }
1106         \__scripture_reference_start_inline:
1107     }
1108 }
1109 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1110 {
1111     \bool_if:NTF \l__scripture_ref_start_newline_bool
1112     { \__scripture_reference_start_newline: }
1113     {
1114         \cs_set_eq:NN \__scripture_temp_newvs: \newvs
1115         \RenewDocumentCommand \newvs { }
1116         { \cs_set_eq:NN \newvs \__scripture_temp_newvs: }
1117         \__scripture_reference_start_inline:
1118     }

```

```

1119     }
1120 }
1121 {
1122     \bool_if:NTF \l__scripture_ref_start_newline_bool
1123     {
1124         \__scripture_reference_start_newline:
1125         \bool_if:NTF \l__scripture_compact_bool
1126         {
1127             \bool_if:NT \g__scripture_mode_vertical_inner_bool
1128             { \__scripture_pilcrow_output: }
1129         }
1130         {
1131             \str_if_eq:VnF
1132             \l__scripture_currenvir_str { poetry }
1133             { \__scripture_pilcrow_output: }
1134         }
1135     }
1136     { \__scripture_reference_start_inline: }
1137 }
1138 \exp_args:No \token_if_eq_meaning:NNF {##1} \begin
1139 {
1140     \tl_clear:N \l__scripture_ref_tl
1141     \str_if_eq:VnF \l__scripture_version_position_str { end }
1142     { \tl_clear:N \l__scripture_version_tl }
1143 }
1144 \peek_analysis_map_break:n
1145 { \__scripture_fix_protrusion:o {##1} }
1146 }
1147 }
1148 {
1149     \peek_analysis_map_inline:n
1150     {
1151         \bool_case:nF
1152         {
1153             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \heading }
1154             { }
1155             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \noindent }
1156             { }
1157             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nohang }
1158             { }
1159             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nopilcrow }
1160             {
1161                 \peek_analysis_map_break:n
1162                 {
1163                     ##1
1164                     \__scripture_fix_protrusion:n { }
1165                 }
1166             }
1167             { \exp_args:No \token_if_eq_meaning_p:NN {##1} \begin }
1168             {
1169                 \bool_if:NF \l__scripture_indent_bool
1170                 {
1171                     \hook_gput_next_code:nn { para / begin }
1172                     {

```

```

1173         \bool_lazy_or:nnT
1174         {
1175             \str_if_eq_p:Vn
1176             \l__scripture_currenvir_str
1177             { midparachap }
1178         }
1179         {
1180             \str_if_eq_p:Vn
1181             \l__scripture_currenvir_str
1182             { scripture }
1183         }
1184         { \para_omit_indent: }
1185     }
1186 }
1187 }
1188 {
1189     \exp_args:No \token_if_eq_meaning_p:NN
1190     {##1} \canonicalheading
1191 }
1192 { }
1193 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \ch }
1194 { }
1195 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newch }
1196 { }
1197 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1198 {
1199     \mode_if_vertical:F
1200     { \__scripture_set_pilcrow_hook: }
1201 }
1202 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1203 {
1204     \mode_if_vertical:F
1205     { \__scripture_set_pilcrow_hook: }
1206 }
1207 }
1208 {
1209     \bool_if:NTF \l__scripture_compact_bool
1210     {
1211         \bool_if:NT \g__scripture_mode_vertical_inner_bool
1212         { \__scripture_pilcrow_output: }
1213     }
1214     {
1215         \str_if_eq:VnF \l__scripture_currenvir_str { poetry }
1216         { \__scripture_pilcrow_output: }
1217     }
1218 }
1219 \peek_analysis_map_break:n
1220 {
1221     \__scripture_fix_protrusion:o {##1}
1222 }
1223 }
1224 }
1225 }
1226 }

```

(End of definition for _scripture_reference_start_peek:.)

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

```

1227 \cs_new_protected:Nn \_scripture_compact_par:
1228 {
1229   \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
1230   \dim_set:Nn \parindent { \l__scripture_parindent_tl }
1231   \peek_remove_spaces:n
1232   {
1233     \peek_analysis_map_inline:n
1234     {
1235       \bool_case:n
1236       {
1237         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \begin }
1238         { \bool_gset_true:N \g__scripture_mode_vertical_inner_bool }
1239         { \str_if_eq_p:Vn \l__scripture_currenvir_str { poetry } }
1240         { }
1241         { \exp_args:No \token_if_letter_p:N {##1} }
1242         { \_scripture_pilcrow_output: }
1243         { \exp_args:No \token_if_active_p:N {##1} }
1244         { \_scripture_pilcrow_output: }
1245         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1246         { \_scripture_set_pilcrow_hook: }
1247         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1248         { \_scripture_set_pilcrow_hook: }
1249       }
1250       \peek_analysis_map_break:n {##1}
1251     }
1252   }
1253 }

```

(End of definition for _scripture_compact_par:.)

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

```

1254 \cs_new_protected:Nn \_scripture_para_noindent:
1255 {
1256   \legacy_if:nTF { @newlist }
1257   {
1258     \_scripture_noindent_saved:
1259     \peek_analysis_map_inline:n
1260     {
1261       \bool_lazy_or:nnTF
1262       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1263       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1264       { \bool_gset_true:N \g__scripture_suppress_next_verse_para_bool }
1265       { \_scripture_pilcrow_output: }
1266       \peek_analysis_map_break:n {##1}
1267     }
1268   }
1269   {
1270     \peek_analysis_map_inline:n
1271     {
1272       \bool_lazy_or:nnTF
1273       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }

```

```

1274         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1275         {
1276             \bool_if:NT \l__scripture_verse_para_bool
1277             { \para_end: }
1278             \__scripture_noindent_saved:
1279             \bool_gset_true:N \g__scripture_suppress_next_verse_para_bool
1280             \__scripture_set_pilcrow_hook:
1281         }
1282         { \__scripture_pilcrow_output: }
1283         \peek_analysis_map_break:n {##1}
1284     }
1285 }
1286 }

```

(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.

```

1287 \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.

```

1288 \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.

```

1289 \cs_new_protected:Nn \__scripture_para_par:
1290 {
1291     \bool_gset_true:N \g__scripture_para_mode_vertical_bool
1292     \dim_set:Nn \parindent { \l__scripture_parindent_tl }
1293     \peek_remove_spaces:n
1294     {
1295         \peek_analysis_map_inline:n
1296         {
1297             \bool_case:n
1298             {
1299                 { \exp_args:No \token_if_letter_p:N {##1} }
1300                 { \__scripture_pilcrow_output: }
1301                 { \exp_args:No \token_if_active_p:N {##1} }
1302                 { \__scripture_pilcrow_output: }
1303                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
1304                 { \__scripture_set_pilcrow_hook: }
1305                 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
1306                 { \__scripture_set_pilcrow_hook: }
1307             }
1308             \peek_analysis_map_break:n {##1}
1309         }
1310     }
1311 }

```

(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.

```

1312 \cs_new_protected:Nn \__scripture_doendpe:
1313 {
1314   \box_gclear:N \g__scripture_post_env_indent_box
1315   \legacy_if_set_true:n { @endpe }
1316   \cs_set_protected:Npn \par
1317   {
1318     \@restorepar
1319     \clubpenalty \@clubpenalty
1320     \everypar { }
1321     \par
1322     \legacy_if_set_false:n { @endpe }
1323     \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
1324     \__scripture_para_par:
1325   }
1326   \everypar
1327   {
1328     \box_gset_to_last:N \g__scripture_post_env_indent_box
1329     \hook_gput_next_code:nn { para / after }
1330     { \box_gclear:N \g__scripture_post_env_indent_box }
1331     \everypar { }
1332     \legacy_if_set_false:n { @endpe }
1333     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1334     \bool_if:NT \l__scripture_verse_para_bool
1335     {
1336       \box_set_wd:Nn
1337       \g__scripture_post_env_indent_box
1338       { \l__scripture_para_indent_tl }
1339     }
1340   }
1341 }

```

(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`.

```

1342 \cs_new_protected:Nn \__scripture_fix_protrusion:n
1343 {
1344   \bool_if:NTF \l__scripture_compact_bool
1345   {
1346     \str_if_eq:VnT \l__scripture_currenvir_str { scripture }
1347     { \cs_if_exist_use:N \leftprotrusion }
1348   }
1349   { \cs_if_exist_use:N \leftprotrusion }
1350   #1
1351 }
1352 \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.

```

1353 \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.

```
1354 \cs_new_protected:Nn \__scripture_red_letter_on:
1355 {
1356   \bool_if:NT \l__scripture_red_letter_bool
1357   {
1358     \bool_gset_true:N \g__scripture_red_letter_active_bool
1359     \exp_args:NV \color_select:n \l__scripture_red_letter_colour_tl
1360     \str_if_eq:VnF \l__scripture_currenvir_str { scripture }
1361     {
1362       \hook_gput_next_code:nn
1363       { env / \l__scripture_currenvir_str / after }
1364       {
1365         \exp_args:NV \color_select:n \l__scripture_red_letter_colour_tl
1366       }
1367     }
1368   }
1369 }
```

(End of definition for __scripture_red_letter_on:.)

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

```
1370 \cs_new_protected:Nn \__scripture_red_letter_off:
1371 {
1372   \bool_if:NT \l__scripture_red_letter_bool
1373   {
1374     \color_select:n { scripture default colour }
1375     \bool_gset_false:N \g__scripture_red_letter_active_bool
1376     \str_if_eq:VnF \l__scripture_currenvir_str { scripture }
1377     {
1378       \hook_gput_next_code:nn
1379       { env / \l__scripture_currenvir_str / after }
1380       {
1381         \color_select:n { scripture default colour }
1382       }
1383     }
1384   }
1385 }
```

(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.

```
1386 \cs_new_protected:Nn \__scripture_select_language:n
1387 {
1388   \tl_set_eq:Nc \l__scripture_language_tl
1389   { l__scripture_#1_language_tl }
1390   \tl_set_eq:Nc \l__scripture_language_variant_tl
1391   { l__scripture_#1_language_variant_tl }
1392   \tl_if_empty:NF \l__scripture_language_tl
1393   {
```

```

1394 \@ifpackageloaded { babel }
1395 {
1396   \selectlanguage { \l__scripture_language_tl }
1397 }
1398 {
1399   \@ifpackageloaded { polyglossia }
1400   {
1401     \iflanguageloaded { \l__scripture_language_tl }
1402     {
1403       \tl_if_empty:NTF \l__scripture_language_variant_tl
1404       {
1405         \selectlanguage { \l__scripture_language_tl }
1406       }
1407       {
1408         \selectlanguage
1409         [ variant = \l__scripture_language_variant_tl ]
1410         { \l__scripture_language_tl }
1411       }
1412     }
1413     {
1414       \msg_error:nnx
1415       { scripture }
1416       { language-error }
1417       { \l__scripture_language_tl }
1418     }
1419   }
1420   {
1421     \msg_warning:nnx
1422     { scripture }
1423     { language-warning }
1424     { \l__scripture_language_tl }
1425   }
1426 }
1427 }
1428 }

```

(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`

```

1429 \box_new:N \l__scripture_chap_box
1430 \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.

```

1431 \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.

```

1432 \dim_new:N \l__scripture_chap_height_dim

```

(End of definition for \l__scripture_chap_height_dim.)

<code>\g__scripture_chap_width_dim</code>	Calculated width of drop chapter number. 1433 <code>\dim_new:N \g__scripture_chap_width_dim</code> <i>(End of definition for \g__scripture_chap_width_dim.)</i>
<code>\g__scripture_chap_par_prevgraf_int</code>	Store the number of lines a chapter paragraph takes up. 1434 <code>\int_new:N \g__scripture_chap_par_prevgraf_int</code> <i>(End of definition for \g__scripture_chap_par_prevgraf_int.)</i>
<code>\l__scripture_chap_X_height_dim</code>	Calculated height of X at current font size. This is used as the height for a drop chapter. 1435 <code>\dim_new:N \l__scripture_chap_X_height_dim</code> <i>(End of definition for \l__scripture_chap_X_height_dim.)</i>
<code>\l__scripture_para_chap_indent_dim</code>	Indent used to center chapters when placed in their own paragraph. 1436 <code>\dim_new:N \l__scripture_para_chap_indent_dim</code> <i>(End of definition for \l__scripture_para_chap_indent_dim.)</i>
<code>\l__scripture_nodrop_chap_coffin</code>	Coffin to place non-drop chapters inside. 1437 <code>\coffin_new:N \l__scripture_nodrop_chap_coffin</code> <i>(End of definition for \l__scripture_nodrop_chap_coffin.)</i>
<code>\l__scripture_nodrop_chap_voffset_dim</code>	Vertical offset to use when setting non-drop chapters. 1438 <code>\dim_new:N \l__scripture_nodrop_chap_voffset_dim</code> <i>(End of definition for \l__scripture_nodrop_chap_voffset_dim.)</i>
<code>__scripture_get_X_height:</code>	Calculate the height of <code>\l__scripture_X_char_tl</code> for vertical alignment of non-drop chapters. 1439 <code>\cs_new_protected:Nn __scripture_get_X_height:</code> 1440 <code>{</code> 1441 <code> \hbox_set:Nn \l__scripture_chap_tmp_box { \l__scripture_X_char_tl }</code> 1442 <code> \dim_set:Nn \l__scripture_chap_X_height_dim</code> 1443 <code> { \box_ht:N \l__scripture_chap_tmp_box }</code> 1444 <code>}</code> <i>(End of definition for __scripture_get_X_height:.)</i>
<code>__scripture_drop_chap_set_up:n</code>	Set up chapter drop cap and calculate its various dimensions. 1445 <code>\cs_new_protected:Nn __scripture_drop_chap_set_up:n</code> 1446 <code>{</code> 1447 <code> \tl_gset:Nn \scripturecurrentchapter {#1}</code> 1448 <code> \tl_gset:Nn \scripturecurrentverse { 1 }</code> 1449 <code> __scripture_get_X_height:</code> 1450 <code> \dim_set:Nn \l__scripture_chap_height_dim</code> 1451 <code> { \baselineskip + \l__scripture_chap_X_height_dim }</code> 1452 <code> \dim_set:Nn \l__scripture_chap_fontsize_dim</code> 1453 <code> {</code> 1454 <code> \fontsize pt *</code> 1455 <code> \dim_ratio:nn</code> 1456 <code> { \l__scripture_chap_height_dim }</code> 1457 <code> { \l__scripture_chap_X_height_dim }</code> 1458 <code> }</code>

```

1458     }
1459     \hbox_set:Nn \l__scripture_chap_tmp_box
1460     {
1461         \l__scripture_chap_font_tl
1462         \fontsize
1463         { \l__scripture_chap_fontsize_dim }
1464         { \l__scripture_chap_fontsize_dim }
1465         \selectfont
1466         \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
1467         { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
1468         \__scripture_select_language:n { chapter }
1469         \__scripture_chap_format:n {#1}
1470     }
1471     \dim_gset:Nn \g__scripture_chap_width_dim
1472     { \box_wd:N \l__scripture_chap_tmp_box }

```

Avoid breaks in the middle of drop chapters.

```

1473     \int_set_eq:NN \l__scripture_clubpenalty_saved_int \clubpenalty
1474     \clubpenalty 10000
1475     \bool_if:NF \l__scripture_verse_para_bool
1476     {
1477         \str_if_eq:VnTF \l__scripture_currenvir_str { poetry }
1478         {
1479             \int_gset_eq:NN
1480             \g__scripture_current_group_level_b_int \currentgrouplevel
1481         }
1482         {
1483             \int_gset:Nn \g__scripture_current_group_level_b_int
1484             { \currentgrouplevel + 1 }
1485         }
1486         \__scripture_chap_par_save_prevgraf_at_group_level:
1487     }
1488 }

```

(End of definition for __scripture_drop_chap_set_up:n.)

__scripture_drop_chap_output: Output a drop chapter.

```

1489 \cs_new_protected:Nn \__scripture_drop_chap_output:
1490 {
1491     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1492     \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
1493     {
1494         \addvspace
1495         {
1496             \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
1497         }
1498     }
1499     \cs_if_eq:NNTF \noindent \__scripture_para_noindent:
1500     {
1501         \legacy_if:nTF { @newlist }
1502         {
1503             \skip_horizontal:n { - \l__scripture_parindent_tl }
1504         }
1505         { \__scripture_noindent_saved: }
1506         \__scripture_pilcrow_output:

```

```

1507     }
1508     { \noindent }
1509     \parshape 3 ~
1510     \dim_eval:n
1511     {
1512         \@totalleftmargin +
1513         \g__scripture_chap_width_dim +
1514         \l__scripture_chap_sep_tl
1515     } ~
1516     \dim_eval:n
1517     {
1518         \linewidth -
1519         \g__scripture_chap_width_dim -
1520         \l__scripture_chap_sep_tl
1521     } ~
1522     \dim_eval:n
1523     {
1524         \@totalleftmargin +
1525         \g__scripture_chap_width_dim +
1526         \l__scripture_chap_sep_tl
1527     } ~
1528     \dim_eval:n
1529     {
1530         \linewidth -
1531         \g__scripture_chap_width_dim -
1532         \l__scripture_chap_sep_tl
1533     } ~
1534     \@totalleftmargin ~
1535     \linewidth
1536     \group_begin:
1537     \hook_use:n { scripture / chap / before }
1538     \box_use:N \l__scripture_chap_box
1539     \hook_use:n { scripture / chap / after }
1540     \group_end:
1541     \bool_if:NTF \l__scripture_chap_show_verse_bool
1542     {
1543         \__scripture_verse_output:n { 1 }
1544     }
1545     { \__scripture_insert_verse_mark: }
1546     \bool_if:NT \l__scripture_verse_para_bool
1547     {
1548         \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
1549         \__scripture_chap_par_save_prevgraf_at_group_level:
1550     }
1551     \hook_gc_clear_next_code:n { scripture / pilcrow }
1552 }

```

(End of definition for __scripture_drop_chap_output:.)

`\__scripture_nodrop_chap:n` Format a non-drop chapter.

```

1553 \cs_new_protected:Nn \__scripture_nodrop_chap:n
1554 {
1555     \tl_gset:Nn \scripturecurrentchapter {#1}
1556     \tl_gset:Nn \scripturecurrentverse { 1 }

```

```

1557 \group_begin:
1558 \bool_if:NT \l__scripture_verse_para_bool
1559 {
1560   \legacy_if:nF { @newlist }
1561   {
1562     \mode_if_vertical:F { \para_end: }
1563   }
1564 }
1565 \bool_if:NTF \l__scripture_chap_show_bool
1566 {
1567   \__scripture_get_X_height:
1568   \hcoffin_set:Nn \l__scripture_nodrop_chap_coffin
1569   {
1570     \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
1571     { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
1572     \l__scripture_chap_font_tl
1573     \__scripture_chap_format:n {#1}
1574     \kern \l__scripture_chap_sep_tl
1575   }
1576   \dim_zero:N \l__scripture_nodrop_chap_voffset_dim
1577   \tl_if_eq:NnT \l__scripture_chapter_valign_tl { t }
1578   {
1579     \dim_set_eq:NN
1580     \l__scripture_nodrop_chap_voffset_dim
1581     \l__scripture_chap_X_height_dim
1582   }
1583   \tl_if_eq:NnT \l__scripture_chapter_valign_tl { vc }
1584   {
1585     \dim_set:Nn \l__scripture_nodrop_chap_voffset_dim
1586     { 0.5 \l__scripture_chap_X_height_dim }
1587   }
1588   \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
1589   {
1590     \addvspace
1591     {
1592       \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
1593     }
1594   }
1595   \mode_if_vertical:F
1596   {
1597     \dim_compare:nNnT { \l__scripture_chap_presep_tl } > { \c_zero_dim }
1598     {
1599       \skip_horizontal:n { \l__scripture_chap_presep_tl }
1600     }
1601   }
1602   \hook_use:n { scripture / chap / before }
1603   \bool_if:NTF \l__scripture_chap_smash_bool
1604   {
1605     \mode_leave_vertical:
1606     \smash
1607     {
1608       \exp_args:NNoo
1609       \coffin_typeset:Nnnnn \l__scripture_nodrop_chap_coffin
1610       { \l__scripture_chapter_align_tl }

```

```

1611         { \l__scripture_chapter_valign_tl }
1612         { Opt }
1613         { \l__scripture_nodrop_chap_voffset_dim }
1614     }
1615 }
1616 {
1617     \exp_args:NNoO
1618     \coffin_typeset:Nnnnn \l__scripture_nodrop_chap_coffin
1619     { \l__scripture_chapter_align_tl }
1620     { \l__scripture_chapter_valign_tl }
1621     { Opt }
1622     { \l__scripture_nodrop_chap_voffset_dim }
1623 }
1624 \nobreak
1625 \skip_horizontal:N \c_zero_skip
1626 \hook_use:n { scripture / chap / after }
1627 \bool_if:NTF \l__scripture_chap_show_verse_bool
1628 {
1629     \__scripture_verse_output:n { 1 }
1630 }
1631 { \__scripture_insert_verse_mark: }
1632 \str_if_eq:VnF \l__scripture_currenvir_str { midparachap }
1633 { \__scripture_pilcrow_output: }
1634 }
1635 {
1636     \hook_use:n { scripture / chap / before }
1637     \hook_use:n { scripture / chap / after }
1638     \vs { 1 }
1639 }
1640 \group_end:
1641 \parshape 1 ~ \@totalleftmargin ~ \linewidth
1642 \hook_gc_clear_next_code:n { scripture / pilcrow }
1643 \peek_remove_spaces:n
1644 {
1645     \peek_analysis_map_inline:n
1646     {
1647         \exp_args:No \token_if_eq_meaning:NNT
1648         {##1}
1649         \canonicalheading
1650         { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
1651         \peek_analysis_map_break:n {##1}
1652     }
1653 }
1654 }

```

(End of definition for __scripture_nodrop_chap:n.)

__scripture_para_chap:n Format a chapter in its own paragraph.

```

1655 \cs_new_protected:Nn \__scripture_para_chap:n
1656 {
1657     \tl_gset:Nn \scripturecurrentchapter {#1}
1658     \tl_gset:Nn \scripturecurrentverse { 1 }
1659     \group_begin:
1660     \skip_zero:N \parskip

```

```

1661 \bool_if:NTF \l__scripture_chap_show_bool
1662 {
1663   \dim_set_eq:NN \l__scripture_para_chap_indent_dim \linewidth
1664   \str_if_eq:VnTF \l__scripture_currenvir_str { scripture }
1665   {
1666     \dim_set:Nn \l__scripture_para_chap_indent_dim
1667       { 0.5 \l__scripture_para_chap_indent_dim }
1668   }
1669   {
1670     \dim_add:Nn \l__scripture_para_chap_indent_dim
1671       { \leftmargin + \rightmargin }
1672     \dim_set:Nn \l__scripture_para_chap_indent_dim
1673       { 0.5 \l__scripture_para_chap_indent_dim }
1674     \dim_sub:Nn \l__scripture_para_chap_indent_dim { \leftmargin }
1675   }
1676   \mode_if_vertical:F { \para_end: }
1677   \addvspace { \l__scripture_chap_para_aboveskip_tl }
1678   \bool_gset_eq:NN \g_tmpa_bool \g__scripture_suppress_next_pilcrow_bool
1679   \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1680   \str_if_eq:VnTF \l__scripture_currenvir_str { midparachap }
1681     { \__scripture_noindent_saved: }
1682     { \noindent }
1683   \bool_gset_eq:NN \g__scripture_suppress_next_pilcrow_bool \g_tmpa_bool
1684   \skip_horizontal:N \l__scripture_para_chap_indent_dim
1685   \hbox_overlap_center:n
1686     {
1687       \hook_use:n { scripture / chap / before }
1688       \group_begin:
1689       \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
1690         { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
1691       \l__scripture_chap_font_tl
1692       \__scripture_select_language:n { chapter }
1693       \__scripture_chap_format:n {#1}
1694       \group_end:
1695       \hook_use:n { scripture / chap / after }
1696     }
1697   \para_end:
1698   \nobreak
1699   \bool_lazy_any:nTF
1700     {
1701       { \str_if_eq_p:Vn \l__scripture_currenvir_str { scripture } }
1702       { \str_if_eq_p:Vn \l__scripture_currenvir_str { midparachap } }
1703       { \str_if_eq_p:Vn \l__scripture_currenvir_str { narrow } }
1704     }
1705     {
1706       \legacy_if_set:nn
1707         { @afterindent } { \l__scripture_chap_para_afterindent_bool }
1708     }
1709     {
1710       \legacy_if_set_true:n { @afterindent }
1711     }
1712   \@afterheading
1713 }
1714 {

```

```

1715     \hook_use:n { scripture / chap / before }
1716     \hook_use:n { scripture / chap / after }
1717   }
1718   \peek_meaning:NTF \c_space_token
1719   {
1720     \addvspace { \l__scripture_chap_para_belowskip_tl }
1721     \group_end:
1722     \hook_use:n { scripture / poetry / pilcrow }
1723   }
1724   {
1725     \addvspace { \l__scripture_chap_para_belowskip_tl }
1726     \cs_if_eq:NNTF \vs \__scripture_poetry_mode_vertical_verse:n
1727     {
1728       \bool_gset_eq:NN
1729         \g_tmpa_bool \g__scripture_suppress_next_pilcrow_bool
1730       \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1731       \noindent
1732       \vs { 1 }
1733       \bool_gset_eq:NN
1734         \g__scripture_suppress_next_pilcrow_bool \g_tmpa_bool
1735     }
1736     {
1737       \str_if_eq:VnT \l__scripture_currenvir_str { midparachap }
1738       { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
1739       \vs { 1 }
1740     }
1741     \group_end:
1742     \hook_use:n { scripture / poetry / pilcrow }
1743   }
1744 }

```

(End of definition for __scripture_para_chap:n.)

__scripture_chap:n Position a drop chapter.

```

1745 \cs_new_protected:Nn \__scripture_chap:n
1746 {
1747   \__scripture_drop_chap_set_up:n {#1}
1748   \hbox_set:Nn \l__scripture_chap_box
1749   {
1750     \box_move_down:nn
1751     { \baselineskip }
1752     {
1753       \hbox_overlap_left:n
1754       {
1755         \box_use:N \l__scripture_chap_tmp_box
1756         \skip_horizontal:N \l__scripture_chap_sep_tl
1757       }
1758     }
1759   }
1760   \box_set_ht:Nn \l__scripture_chap_box { \c_zero_dim }
1761   \box_set_dp:Nn \l__scripture_chap_box { \c_zero_dim }
1762   \bool_if:NTF \l__scripture_mid_para_chap_bool
1763   { \__scripture_mid_para_drop_chap_output: }
1764   {

```

```

1765     \bool_if:NTF \l__scripture_verse_para_bool
1766     { \para_end: }
1767     { \par }
1768     \__scripture_drop_chap_output:
1769     \str_if_eq:VnF \l__scripture_currenvir_str { midparachap }
1770     { \__scripture_pilcrow_output: }
1771   }
1772   \int_gset_eq:NN \g__scripture_current_group_level_a_int \currentgrouplevel
1773   \__scripture_reset_par_at_group_level:
1774 }

```

(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.

```

1775 \int_new:N \g__scripture_current_group_level_a_int
1776 \int_new:N \g__scripture_current_group_level_b_int
1777 \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.

```

1778 \cs_new_protected:Nn \__scripture_reset_par_at_group_level:
1779 {
1780   \hook_gput_next_code:nn { para / before }
1781   {
1782     \int_compare:nNnTF
1783     { \g__scripture_current_group_level_a_int } = { \currentgrouplevel }
1784     {
1785       \legacy_if:nF { @newlist }
1786       {
1787         \int_compare:nNnT
1788         { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
1789         {
1790           \dim_compare:nNnT { \parskip } < { \baselineskip }
1791           {
1792             \dim_compare:nNnT
1793             {
1794               \g__scripture_chap_width_dim +
1795               \l__scripture_chap_sep_tl
1796             }
1797             >
1798             { \parindent }
1799             {
1800               \hook_gput_next_code:nn { para / begin }
1801               {
1802                 \para_omit_indent:
1803                 \skip_horizontal:n
1804                 {
1805                   \g__scripture_chap_width_dim +
1806                   \l__scripture_chap_sep_tl
1807                 }

```



```

1808         }
1809     }
1810 }
1811 }
1812 \int_gzero:N \g__scripture_chap_par_prevgraf_int
1813 }
1814 \parshape 1 ~ \@totalleftmargin ~ \linewidth
1815 }
1816 { \__scripture_reset_par_at_group_level: }
1817 }
1818 }

```

(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.

```

1819 \cs_new_protected:Nn \__scripture_chap_par_save_prevgraf_at_group_level:
1820 {
1821   \hook_gput_next_code:nn { para / after }
1822   {
1823     \int_compare:nNnTF
1824       { \g__scripture_current_group_level_b_int } = { \currentgrouplevel }
1825       {
1826         \int_gset_eq:NN \g__scripture_chap_par_prevgraf_int \prevgraf
1827         \int_compare:nNnT
1828           { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
1829           { \penalty 10000 }
1830         \hook_gput_next_code:nn { para / before }
1831         {
1832           \int_set_eq:NN
1833             \clubpenalty \l__scripture_clubpenalty_saved_int
1834         }
1835       }
1836       { \__scripture_chap_par_save_prevgraf_at_group_level: }
1837   }
1838 }

```

(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.

```

1839 \cs_new_protected:Nn \__scripture_zero_chap_par_prevgraf_at_group_level:
1840 {
1841   \hook_gput_next_code:nn { para / after }
1842   {
1843     \int_compare:nNnTF
1844       { \g__scripture_current_group_level_c_int } = { \currentgrouplevel }
1845       {
1846         \hook_gput_next_code:nn { para / after }
1847         { \int_gzero:N \g__scripture_chap_par_prevgraf_int }
1848       }
1849       { \__scripture_zero_chap_par_prevgraf_at_group_level: }
1850   }
1851 }

```

(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. 1852 <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. 1853 <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. 1854 <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. 1855 <code>\cs_new_protected:Nn __scripture_verse_output:n</code> 1856 <code>{</code> 1857 <code>\tl_gset:Nn \scripturecurrentverse {#1}</code> 1858 <code>\legacy_if:nTF { @endpe }</code> 1859 <code>{ \bool_set_false:N \l__scripture_verse_par_start_bool }</code> 1860 <code>{ \bool_set_true:N \l__scripture_verse_par_start_bool }</code> 1861 <code>\mode_if_vertical:TF</code> 1862 <code>{ \mode_leave_vertical: }</code> 1863 <code>{</code> 1864 <code>\bool_set_false:N \l__scripture_verse_par_start_bool</code> 1865 <code>\skip_if_eq:nnF { \l__scripture_verse_presep_tl } { \c_zero_skip }</code> 1866 <code>{</code> 1867 <code>\tl_if_eq:nnF {#1} { 1 }</code> 1868 <code>{</code> 1869 <code>\skip_horizontal:n { \l__scripture_verse_presep_tl }</code> 1870 <code>}</code> 1871 <code>}</code> 1872 <code>}</code> 1873 <code>\bool_if:NT \l__scripture_no_verse_first_bool</code> 1874 <code>{</code> 1875 <code>\bool_set_false:N \l__scripture_verse_par_start_bool</code> 1876 <code>\bool_set_false:N \l__scripture_no_verse_first_bool</code> 1877 <code>}</code> 1878 <code>\group_begin:</code> 1879 <code>\bool_if:NT \l__scripture_verse_para_bool</code> 1880 <code>{</code> 1881 <code>\box_if_empty:NF \g__scripture_post_env_indent_box</code> 1882 <code>{ \box_use_drop:N \g__scripture_post_env_indent_box }</code> 1883 <code>}</code> 1884 <code>\hook_use:n { scripture / verse / before }</code> 1885 <code>\bool_if:NT \l__scripture_verse_show_bool</code> 1886 <code>{</code> 1887 <code>\group_begin:</code> 1888 <code>\l__scripture_verse_font_tl</code> 1889 <code>\str_if_eq:VnF \l__scripture_verse_colour_tl { . }</code> 1890 <code>{ \exp_args:NV \color_select:n \l__scripture_verse_colour_tl }</code> 1891 <code>__scripture_select_language:n { verse }</code>

```

1892 \bool_lazy_all:nTF
1893 {
1894   { \l__scripture_verse_first_bool }
1895   { \l__scripture_verse_par_start_bool }
1896   {
1897     \bool_not_p:n
1898     {
1899       \str_if_eq_p:Vn \l__scripture_currenvir_str { poetry }
1900     }
1901   }
1902 }
1903 {
1904   \__scripture_verse_first_format:n {#1}
1905   \group_end:
1906   \kern \l__scripture_verse_first_sep_tl
1907 }
1908 {
1909   \__scripture_verse_format:n {#1}
1910   \group_end:
1911   \kern \l__scripture_verse_sep_tl
1912 }
1913 \nobreak
1914 \skip_horizontal:N \c_zero_skip
1915 }
1916 \__scripture_insert_verse_mark:
1917 \hook_use:n { scripture / verse / after }
1918 \hook_use:n { scripture / pilcrow }
1919 \group_end:
1920 }

```

(End of definition for __scripture_verse_output:n.)

__scripture_verse_compact_output:n Output a formatted verse.

```

1921 \cs_new_protected:Nn \__scripture_verse_compact_output:n
1922 {
1923   \mode_if_vertical:T
1924   { \__scripture_set_pilcrow_hook: }
1925   \__scripture_verse_output:n {#1}
1926 }

```

(End of definition for __scripture_verse_compact_output:n.)

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

```

1927 \cs_new_protected:Nn \__scripture_verse_para_output:n
1928 {
1929   \mode_if_vertical:T
1930   { \__scripture_set_pilcrow_hook: }
1931   \hook_if_empty:nT { scripture / pilcrow }
1932   {
1933     \dim_set:Nn \parindent { \l__scripture_para_indent_tl }
1934   }
1935   \bool_if:NTF \g__scripture_suppress_next_verse_para_bool
1936   { \bool_gset_false:N \g__scripture_suppress_next_verse_para_bool }
1937   {
1938     \legacy_if:nF { @newlist }

```

```

1939         {
1940             \mode_if_vertical:F
1941             {
1942                 \para_end:
1943                 \parshape 1 ~ \@totalleftmargin ~ \linewidth
1944             }
1945         }
1946     }
1947     \__scripture_verse_output:n {#1}
1948 }

```

(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.

```

1949 \cs_new_protected:Nn \__scripture_hanging_verse_para:n
1950 {
1951     \mode_if_vertical:T
1952     { \__scripture_set_pilcrow_hook: }
1953     \bool_if:NF \g__scripture_suppress_next_verse_para_bool
1954     {
1955         \legacy_if:nF { @newlist }
1956         {
1957             \mode_if_vertical:F
1958             {
1959                 \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
1960                 \hook_gput_next_code:nn { para / begin }
1961                 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
1962                 \par
1963                 \__scripture_noindent_saved:
1964             }
1965         }
1966     }
1967     \bool_gset_false:N \g__scripture_suppress_next_verse_para_bool
1968     \__scripture_hanging_verse:n {#1}
1969 }

```

(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.

```

1970 \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.

```

1971 \cs_new_protected:Nn \__scripture_set_pilcrow_hook:
1972 {
1973     \hook_if_empty:nT { scripture / pilcrow }
1974     {
1975         \hook_gput_next_code:nn
1976         { scripture / pilcrow } { \__scripture_pilcrow_output: }
1977     }
1978 }

```

(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.

```

1979 \cs_new_protected:Nn \__scripture_pilcrow_output:
1980 {
1981   \bool_gset_false:N \g__scripture_para_mode_vertical_bool
1982   \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
1983   \bool_if:NT \l__scripture_pilcrow_bool
1984   {
1985     \bool_lazy_or:nnT
1986     { \l__scripture_verse_para_bool } { \l__scripture_compact_bool }
1987     {
1988       \bool_if:NTF \g__scripture_suppress_next_pilcrow_bool
1989       { \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool }
1990       {
1991         \P
1992         \kern \l__scripture_pilcrow_sep_tl
1993         \nobreak
1994         \skip_horizontal:N \c_zero_skip
1995       }
1996     }
1997   }
1998 }
```

(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.

```

1999 \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.

```

2000 \cs_new_protected:Nn \__scripture_calc_final_line_length:
2001 {
2002   \skip_set:Nn \abovedisplayshortskip { - \baselineskip }
2003   \skip_set_eq:NN \abovedisplayskip \abovedisplayshortskip
2004   \skip_zero:N \belowdisplayshortskip
2005   \skip_zero:N \belowdisplayskip
2006   $$
2007   \hook_use:n { para / after }
2008   \dim_compare:nNnT { \predisplaysize } > { \c_zero_dim }
2009   {
2010     \dim_gset:Nn \g__scripture_final_line_dim
2011     { \predisplaysize - 2em }
2012   }
2013   \cs_gset_nopar:Nx \__scripture_reset_spacing:
2014   {
2015     \dim_set:Nn \predisplaysize { \dim_use:N \predisplaysize }
2016     \int_set:Nn \prevgraf { \int_use:N \prevgraf }
2017     \int_gset:Nn \g__scripture_chap_par_prevgraf_int
2018     { \int_use:N \g__scripture_chap_par_prevgraf_int }

```

```

2019     }
2020     $$
2021     \__scripture_reset_spacing:
2022     \dim_compare:nNnT { \g__scripture_final_line_dim } > { \paperwidth }
2023     {
2024         \dim_gset:Nn \g__scripture_final_line_dim
2025         { \@totalleftmargin + \linewidth }
2026     }
2027     \dim_compare:nNnT { \g__scripture_final_line_dim } > { \c_zero_dim - 1sp }
2028     {
2029         \skip_vertical:n { - \baselineskip - \parskip }
2030     }
2031 }

```

(End of definition for __scripture_calc_final_line_length:.)

3.5.8 The **scripture** environment

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

```

2032 \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.

```

2033 \tl_new:N \l__scripture_ref_tl

```

(End of definition for \l__scripture_ref_tl.)

`\__scripture_added:n` Mark text as added.

```

2034 \cs_new_protected_nopar:Nn \__scripture_added:n
2035 {
2036     \group_begin:
2037     \l__scripture_added_font_tl
2038     \__scripture_added_format:n {#1}
2039     \group_end:
2040 }

```

(End of definition for __scripture_added:n.)

`\__scripture_canonicalheading:n` Output a canonical heading.

```

2041 \cs_new_protected_nopar:Nn \__scripture_canonicalheading:n
2042 {
2043     \str_if_eq:VnTF \l__scripture_currenvir_str { scripture }
2044     {
2045         \parshape 1 ~ \@totalleftmargin ~ \linewidth
2046         \group_begin:
2047         \legacy_if:nTF { @newlist }
2048         {
2049             \skip_horizontal:n { - \parindent }
2050         }
2051         {
2052             \__scripture_list_para_end:
2053             \addvspace { \l__scripture_canonicalheading_aboveskip_tl }
2054             \__scripture_noindent_saved:
2055         }

```

```

2056 \l__scripture_canonicalheading_align_tl
2057 \l__scripture_canonicalheading_font_tl
2058 \__scripture_canonicalheading_format:n {#1}
2059 \__scripture_list_para_end:
2060 \group_end:
2061 \bool_if:NTF \l__scripture_verse_para_bool
2062 {
2063   \cs_set_eq:NN \noindent \__scripture_para_noindent:
2064 }
2065 {
2066   \cs_set_eq:NN \noindent \__scripture_noindent_saved:
2067 }
2068 \legacy_if_set:nn
2069 { @afterindent } { \l__scripture_canonicalheading_afterindent_bool }
2070 \bool_if:NF \l__scripture_canonicalheading_afterindent_bool
2071 {
2072   \hook_gput_next_code:nn { para / before }
2073   { \para_omit_indent: }
2074 }
2075 \nobreak
2076 \@afterheading
2077 \addvspace { \l__scripture_canonicalheading_belowskip_tl }
2078 }
2079 {
2080   \msg_error:nnx
2081   { scripture }
2082   { heading-error }
2083   { \l__scripture_currenvir_str }
2084 }
2085 }

```

(End of definition for __scripture_canonicalheading:n.)

__scripture_heading:n Output a non-canonical heading.

```

2086 \cs_new_protected_nopar:Nn \__scripture_heading:n
2087 {
2088   \str_if_eq:VnTF \l__scripture_currenvir_str { scripture }
2089   {
2090     \bool_if:NTF \l__scripture_heading_show_bool
2091     {
2092       \parshape 1 ~ \@totalleftmargin ~ \linewidth
2093       \group_begin:
2094       \bool_set_false:N \l__scripture_pilcrow_bool
2095       \legacy_if:nTF { @newlist }
2096       {
2097         \skip_horizontal:n { - \parindent }
2098       }
2099       {
2100         \__scripture_list_para_end:
2101         \addvspace { \l__scripture_heading_aboveskip_tl }
2102         \__scripture_noindent_saved:
2103       }
2104       \l__scripture_heading_align_tl
2105       \l__scripture_heading_font_tl

```

```

2106      \__scripture_heading_format:n {#1}
2107      \mark_insert:nn { scripture / heading } {#1}
2108      \__scripture_list_para_end:
2109      \group_end:
2110      \bool_if:NTF \l__scripture_verse_para_bool
2111      {
2112          \cs_set_eq:NN \noindent \__scripture_para_noindent:
2113      }
2114      {
2115          \cs_set_eq:NN \noindent \__scripture_noindent_saved:
2116      }
2117      \legacy_if_set:nn
2118      { @afterindent } { \l__scripture_heading_afterindent_bool }
2119      \bool_if:NF \l__scripture_heading_afterindent_bool
2120      {
2121          \hook_gput_next_code:nn { para / before }
2122          { \para_omit_indent: }
2123      }
2124      \nobreak
2125      \@afterheading
2126      \addvspace { \l__scripture_heading_belowskip_tl }
2127      }
2128      {
2129          \mark_insert:nn { scripture / heading } {#1}
2130          \ignorespaces
2131      }
2132      }
2133      {
2134          \msg_error:nnx
2135          { scripture }
2136          { heading-error }
2137          { \l__scripture_currenvir_str }
2138      }
2139      }

```

(End of definition for __scripture_heading:n.)

__scripture_name:n Output divine name.

```

2140      \cs_new_protected_nopar:Nn \__scripture_name:n
2141      {
2142          \group_begin:
2143          \l__scripture_name_font_tl
2144          \__scripture_name_format:n {#1}
2145          \group_end:
2146      }

```

(End of definition for __scripture_name:n.)

__scripture_no_first_verse: Suppress special formatting for first verse if active.

```

2147      \cs_new_protected_nopar:Nn \__scripture_no_first_verse:
2148      {
2149          \bool_set_true:N \l__scripture_no_verse_first_bool
2150      }

```

(End of definition for __scripture_no_first_verse:.)

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

```

2151 \cs_new_protected:Nn \__scripture_ch_output:nn
2152 {
2153   \bool_if:NTF \l__scripture_chap_show_bool
2154   {
2155     \bool_if:NTF \l__scripture_chap_drop_bool
2156     {
2157       \__scripture_chap:n {#2}
2158       \bool_lazy_and:nnTF
2159       {#1}
2160       {
2161         \bool_not_p:n { \l__scripture_compact_bool }
2162       }
2163       {
2164         \hook_gput_next_code:nn { para / after }
2165         {
2166           \int_gzero:N \g__scripture_chap_par_prevgraf_int
2167           \int_gzero:N \g__scripture_mid_para_chap_prevgraf_int
2168         }
2169       }
2170       {
2171         \hook_gput_next_code:nn { para / after }
2172         {
2173           \int_gset_eq:NN
2174           \g__scripture_mid_para_chap_prevgraf_int \prevgraf
2175         }
2176       }
2177     }
2178     {
2179       \bool_if:NTF \l__scripture_chap_para_bool
2180       { \__scripture_para_chap:n {#2} }
2181       { \__scripture_nodrop_chap:n {#2} }
2182     }
2183   }
2184   {
2185     \bool_if:NTF \l__scripture_chap_para_bool
2186     { \__scripture_para_chap:n {#2} }
2187     { \__scripture_nodrop_chap:n {#2} }
2188   }
2189 }

```

(End of definition for __scripture_ch_output:nn.)

`\__scripture_begin:nn` Begin a scripture quotation.

```

2190 \cs_new_protected:Nn \__scripture_begin:nn
2191 {
2192   \cs_set_eq:NN \__scripture_noindent_saved: \noindent
2193   \bool_if:NT \l__scripture_active_bool
2194   {
2195     \msg_error:nn { scripture } { nested-environment }
2196   }
2197   \bool_set_true:N \l__scripture_active_bool
2198   \tl_if_novalue:nF {#1}
2199   { \tl_set:Nn \l__scripture_ref_tl {#1} }

```

```

2200 \str_set:Nn \l__scripture_currenvir_str { scripture }
2201 \bool_gset_false:N \g__scripture_first_verse_set_bool
2202 \int_gset:Nn \g__scripture_chapter_int { 1 }
2203 \int_gset:Nn \g__scripture_verse_int { 1 }
2204 \tl_gclear:N \g__scripture_book_tl
2205 \tl_if_novalue:nF {#2}
2206 {
2207   \keys_set:nn { scripture } {#2}
2208 }
2209 \tl_gset:NV \scripturecurrentbook \g__scripture_book_tl
2210 \tl_gset:Ne \scripturecurrentchapter
2211 { \int_use:N \g__scripture_chapter_int }
2212 \tl_gset:Ne \scripturecurrentverse
2213 { \int_use:N \g__scripture_verse_int }
2214 \l__scripture_before_tl
2215 \__scripture_select_language:n { main }
2216 \int_gzero:N \g__scripture_chap_par_prevgraf_int
Empty \everydisplay to ensure it isn't interfering with spacing.
2217 \everydisplay { }
2218 \int_set:Nn \predisplaypenalty { 10000 }
2219 \int_set_eq:NN \postdisplaypenalty \@lowpenalty
2220 \int_set:Nn \@beginparpenalty { - \@lowpenalty }
2221 \int_set_eq:NN \@endparpenalty \@lowpenalty
2222 \bool_gset_false:N \g__scripture_poetry_end_bool
2223 \cs_set_eq:NN \extraskip \__scripture_extra_skip:
2224 \cs_set_eq:NN \redletteron \__scripture_red_letter_on:
2225 \cs_set_eq:NN \redletteroff \__scripture_red_letter_off:
2226 \cs_set_protected:Npn \nopilcrow
2227 { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2228 \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
2229 \bool_gset_false:N \g__scripture_suppress_next_verse_para_bool
2230 \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
2231 \box_gclear:N \g__scripture_post_env_indent_box
2232 \skip_gzero:N \g__scripture_prev_inner_below_skip
2233 \bool_if:NT \l__scripture_chap_para_bool
2234 { \bool_set_false:N \l__scripture_chap_drop_bool }
2235 \DeclareDocumentCommand { \ch } { s o m }
2236 {
2237   \tl_if_novalue:nF {##2}
2238   {
2239     \tl_set:Nn \l__scripture_X_char_tl {##2}
2240   }
2241   \__scripture_ch_output:nn {##1} {##3}
2242 }
2243 \cs_set_eq:NN \vs \__scripture_verse_output:n
2244 \DeclareDocumentCommand { \newch } { s o }
2245 {
2246   \bool_if:NTF \g__scripture_first_verse_set_bool
2247   {
2248     \int_gincr:N \g__scripture_chapter_int
2249     \int_gset:Nn \g__scripture_verse_int { 1 }
2250   }
2251   { \bool_gset_true:N \g__scripture_first_verse_set_bool }
2252   \bool_if:nTF {##1}

```

```

2253     {
2254         \ch* [ ##2 ] { \int_use:N \g__scripture_chapter_int }
2255     }
2256     {
2257         \ch [ ##2 ] { \int_use:N \g__scripture_chapter_int }
2258     }
2259 }
2260 \DeclareDocumentCommand { \newvs } { }
2261 {
2262     \bool_if:NTF \g__scripture_first_verse_set_bool
2263     { \int_gincr:N \g__scripture_verse_int }
2264     { \bool_gset_true:N \g__scripture_first_verse_set_bool }
2265     \vs { \int_use:N \g__scripture_verse_int }
2266 }
2267 \cs_set_eq:NN \added \__scripture_added:n
2268 \cs_set_eq:NN \heading \__scripture_heading:n
2269 \bool_if:NTF \l__scripture_compact_bool
2270 {
2271     \cs_set_protected:Npn \canonicalheading ##1
2272     {
2273         \__scripture_set_pilcrow_hook:
2274         \hook_use:n { scripture / pilcrow }
2275         ##1
2276     }
2277 }
2278 { \cs_set_eq:NN \canonicalheading \__scripture_canonicalheading:n }
2279 \cs_set_eq:NN \name \__scripture_name:n
2280 \cs_set_protected_nopar:Npn \LORD { \name { Lord } }
2281 \cs_set_protected_nopar:Npn \GOD { \name { God } }
2282 \cs_set_eq:NN \nofirstverse \__scripture_no_first_verse:
2283 \cs_set_protected_nopar:Npn \textright ##1
2284 { \__scripture_text_right:nn { \l__scripture_text_right_sep_tl } {##1} }
2285 \hook_gclear_next_code:n { para / before }
2286 \hook_gclear_next_code:n { scripture / pilcrow }
2287 \hook_gclear_next_code:n { scripture / poetry / pilcrow }
2288 \hook_gput_code:nnn { cmd / @makefntext / before } { scripture }
2289 { \cs_set_eq:NN \noindent \__scripture_noindent_saved: }
2290 \bool_if:NTF \l__scripture_inline_bool
2291 { \l__scripture_inline_font_tl }
2292 { \l__scripture_font_tl }
2293 \skip_if_eq:nnF { \l__scripture_verse_presep_tl } { \c_zero_skip }
2294 {
2295     \bool_set_true:N \l__scripture_verse_first_bool
2296     \cs_set_eq:NN
2297         \__scripture_verse_first_format:n \__scripture_verse_format:n
2298 }
2299 \tl_set:Ne \l__scripture_parindent_tl
2300 { \dim_eval:n { \l__scripture_parindent_tl } }
2301 \exp_args:NnV
2302 \color_set:nn { scripture default colour } \l__scripture_colour_tl
2303 \str_if_eq:VnF \l__scripture_colour_tl { . }
2304 { \color_select:n { scripture default colour } }
2305 \bool_if:NTF \l__scripture_inline_bool
2306 { \l__scripture_inline_begin_tl }

```

```

2307 {
2308   \legacy_if:nTF { @newlist }
2309   { \dim_set_eq:NN \l__scripture_outer_itemindent_dim \itemindent }
2310   { \dim_zero:N \l__scripture_outer_itemindent_dim }
2311 \list { }
2312 {
2313   \dim_set:Nn \leftmargin { \l__scripture_leftmargin_tl }
2314   \dim_set:Nn \rightmargin { \l__scripture_rightmargin_tl }
2315   \dim_set:Nn \itemindent
2316   {
2317     \l__scripture_parindent_tl - \l__scripture_outer_itemindent_dim
2318   }
2319   \dim_set:Nn \listparindent { \l__scripture_parindent_tl }
2320   \skip_zero:N \partopsep
2321   \skip_set:Nn \parsep { \l__scripture_parskip_tl }
2322   \dim_compare:nNnTF { \parskip } > { \l__scripture_aboveskip_tl }
2323   { \skip_zero:N \topsep }
2324   {
2325     \skip_set:Nn \topsep
2326     {
2327       \__scripture_skip_diff:NN
2328       \l__scripture_aboveskip_tl \parskip
2329     }
2330   }
2331   \__scripture_setup_list_noindent:n { \l__scripture_parindent_tl }
2332   \bool_if:NF \l__scripture_indent_bool
2333   {
2334     \hook_gput_next_code:nn { para / begin }
2335     {
2336       \bool_if:NTF \l__scripture_compact_bool
2337       { \para_omit_indent: }
2338       {
2339         \bool_lazy_any:nT
2340         {
2341           {
2342             \str_if_eq_p:Vn
2343             \l__scripture_currenvir_str { midparachap }
2344           }
2345           {
2346             \str_if_eq_p:Vn
2347             \l__scripture_currenvir_str { narrow }
2348           }
2349           {
2350             \str_if_eq_p:Vn
2351             \l__scripture_currenvir_str { scripture }
2352           }
2353         }
2354         {
2355           \legacy_if:nTF { @nparlist }
2356           {
2357             \bool_lazy_and:nnF
2358             {
2359               \str_if_eq_p:Vn
2360               \l__scripture_currenvir_str { narrow }

```

```

2361     }
2362     {
2363         \str_if_eq_p:Vn
2364         \l__scripture_ref_position_str { start }
2365     }
2366     { \noindent }
2367 }
2368 { \para_omit_indent: }
2369 }
2370 }
2371 }
2372 }
2373 }
2374 \cs_set_eq:NN \__scripture_list_para_end: \@par
2375 \item
2376 \relax
2377 }
2378 }

```

(End of definition for __scripture_begin:nn.)

__scripture_end: End a scripture quotation with a reference.

```

2379 \cs_new_protected:Nn \__scripture_end:
2380 {
2381     \str_if_eq:VnT \l__scripture_ref_position_str { start }
2382     {
2383         \tl_clear:N \l__scripture_ref_tl
2384         \str_if_eq:VnT \l__scripture_version_position_str { withref }
2385         { \tl_clear:N \l__scripture_version_tl }
2386     }
2387     \cs_set_eq:NN \noindent \__scripture_noindent_saved:
2388     \bool_if:NTF \l__scripture_inline_bool
2389     {
2390         \unskip
2391         \__scripture_red_letter_off:
2392         \l__scripture_inline_end_tl
2393         \bool_lazy_or:nnTF
2394         {
2395             \tl_if_novalue_p:n { \l__scripture_ref_tl }
2396         }
2397         { \tl_if_empty_p:N \l__scripture_ref_tl }
2398         {
2399             \tl_if_empty:NF \l__scripture_version_tl
2400             {
2401                 \skip_horizontal:N \l__scripture_ref_sep_tl
2402                 \__scripture_format_full_ref:nn { } { \l__scripture_version_tl }
2403             }
2404         }
2405         {
2406             \skip_horizontal:N \l__scripture_ref_sep_tl
2407             \__scripture_format_full_ref:nn
2408             { \l__scripture_ref_tl } { \l__scripture_version_tl }
2409         }
2410     }

```

```

2411 {
2412   \__scripture_red_letter_off:
2413   \skip_zero:N \parskip
2414   \int_set:Nn \postdisplaypenalty { 10000 }
2415   \__scripture_calc_final_line_length:
2416   \dim_gsub:Nn \g__scripture_final_line_dim { \@totalleftmargin }
2417   \dim_compare:nNnTF { \@outerparskip } > { \l__scripture_belowskip_tl }
2418     { \skip_zero:N \@topsepadd }
2419     {
2420       \skip_set:Nn \@topsepadd
2421       {
2422         \__scripture_skip_diff:NN
2423         \l__scripture_belowskip_tl \@outerparskip
2424       }
2425     }

```

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

```

2426   \int_compare:nNnT { \prevgraf } = { \c_zero_int }
2427   {
2428     \skip_vertical:n { - \g__scripture_prev_inner_below_skip }
2429   }
2430   \hook_gclear_next_code:n { para / before }
2431   \hook_gclear_next_code:n { para / after }
2432   \noindent
2433   \bool_lazy_or:nnTF
2434   {
2435     \tl_if_novalue_p:n { \l__scripture_ref_tl }
2436   }
2437   { \tl_if_empty_p:N \l__scripture_ref_tl }
2438   {
2439     \tl_if_empty:NNTF \l__scripture_version_tl
2440     { \hbox:n { } }
2441     { \__scripture_reference:n { } }
2442   }
2443   {
2444     \__scripture_reference:n { \l__scripture_ref_tl }
2445   }
2446   \parshape 1 ~ \@totalleftmargin ~ \linewidth

```

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

```

2447   \legacy_if_set_false:n { @nparlist }
2448   \endlist
2449 }
2450 \hook_gremove_code:nn { cmd / @makefnstext / before } { scripture }
2451 \cs_set_eq:NN \noindent \__scripture_noindent_saved:
2452 \l__scripture_after_tl
2453 }

```

(End of definition for __scripture_end:.)

3.6 Scripture internal environments

`\__scripture_compact_end_inner:` End a compact inner environment

```

2454 \cs_new_protected:Nn \__scripture_compact_end_inner:

```

```

2455 {
2456   \hook_gclear_next_code:n { para / before }
2457   \hook_gclear_next_code:n { para / begin }
2458   \bool_gset_false:N \g__scripture_mode_vertical_inner_bool
2459   \str_if_eq:VnT \l__scripture_ref_position_str { start }
2460   {
2461     \tl_gclear:N \l__scripture_ref_tl
2462   }
2463   \bool_lazy_and:nnT
2464   { \l__scripture_verse_para_bool } { \l__scripture_chap_drop_bool }
2465   {
2466     \cs_set_eq:NN \par \para_end:
2467     \__scripture_calc_final_line_length:
2468     \int_gset_eq:NN
2469     \g__scripture_mid_para_chap_prevgraf_int
2470     \g__scripture_chap_par_prevgraf_int
2471     \__scripture_noindent_saved:
2472     \bool_lazy_and:nnTF
2473     { \bool_if_p:N \l__scripture_compact_bool }
2474     {
2475       \int_compare_p:nNn
2476       { \g__scripture_mid_para_chap_prevgraf_int } = { \c_one_int }
2477     }
2478     {
2479       \hook_gput_next_code:nn
2480       { env / \l__scripture_currenvir_str / after }
2481       {
2482         \parshape 3 ~
2483         \@totalleftmargin ~ \linewidth ~
2484         \dim_eval:n
2485         {
2486           \@totalleftmargin +
2487           \g__scripture_chap_width_dim +
2488           \l__scripture_chap_sep_tl
2489         } ~
2490         \dim_eval:n
2491         {
2492           \linewidth -
2493           \g__scripture_chap_width_dim -
2494           \l__scripture_chap_sep_tl
2495         } ~
2496         \@totalleftmargin ~ \linewidth
2497       }
2498       \hbox_to_wd:nn
2499       {
2500         \g__scripture_final_line_dim -
2501         \g__scripture_chap_width_dim -
2502         \l__scripture_chap_sep_tl
2503       }
2504       { }
2505     }
2506     {
2507       \hbox_to_wd:nn { \g__scripture_final_line_dim } { }
2508     }

```

```

2509     \c_space_token
2510     \kern 0pt
2511     \cs_set_eq:NN \par \relax
2512     \int_gset:Nn \g__scripture_current_group_level_a_int
2513       { \currentgrouplevel - \c_one_int }
2514     \__scripture_reset_par_at_group_level:
2515   }
2516   \unskip
2517 }

```

(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.

```

2518 \cs_new_protected:Nn \__scripture_center_begin:n
2519 {
2520   \bool_if:NT \l__scripture_active_inner_bool
2521   {
2522     \msg_error:nn { scripture } { nested-environment }
2523   }
2524   \bool_set_true:N \l__scripture_active_inner_bool
2525   \str_set:Nn \l__scripture_currenvir_str { center }
2526   \tl_if_novalue:nF {#1}
2527   {
2528     \keys_set:nn
2529       { scripture / center }
2530       {#1}
2531   }
2532   \legacy_if:nTF { @newlist }
2533   {
2534     \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
2535   }
2536   { \dim_zero:N \l__scripture_outer_itemindent_dim }
2537   \cs_set_eq:NN \par \para_end:
2538   \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
2539   \list { }
2540   {
2541     \bool_lazy_or:nnF
2542       { \mode_if_vertical_p: }
2543       { \g__scripture_para_mode_vertical_bool }
2544       { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2545     \dim_zero:N \labelsep
2546     \dim_set:Nn \leftmargin { \l__scripture_center_leftmargin_tl }
2547     \dim_set:Nn \rightmargin { \l__scripture_center_rightmargin_tl }
2548     \dim_set:Nn \itemindent { - \l__scripture_outer_itemindent_dim }
2549     \dim_zero:N \listparindent
2550     \skip_set_eq:NN \parsep \parskip
2551     \skip_zero:N \partopsep
2552     \dim_compare:nNnTF
2553       { \parskip } > { \l__scripture_center_aboveskip_tl }
2554       { \skip_zero:N \topsep }
2555       {
2556         \skip_set:Nn \topsep

```



```

2557         {
2558             \__scripture_skip_diff:NN
2559             \l__scripture_center_aboveskip_tl \parskip
2560         }
2561     }
2562     \__scripture_setup_list_noindent:n { \c_zero_dim }
2563 }
2564 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2565 { \addvspace { \baselineskip } }
2566 \int_gzero:N \g__scripture_chap_par_prevgraf_int
2567 \centering
2568 \bool_if:NT \l__scripture_verse_para_bool
2569 {
2570     \cs_set_eq:NN \par \__scripture_para_par:
2571     \cs_set_protected:Npn \\\
2572     {
2573         \mode_if_horizontal:TF
2574         { \unskip }
2575         { \@nolnerr }
2576     }
2577 }
2578 \item
2579 \relax
2580 }

```

(End of definition for __scripture_center_begin:n.)

__scripture_center_end: End a centred section in a scripture quotation.

```

2581 \cs_new_protected:Nn \__scripture_center_end:
2582 {
2583     \cs_set_eq:NN \para_end:
2584     \dim_gset:Nn \g__scripture_final_line_dim
2585     { \@totalleftmargin + \linewidth }
2586     \dim_compare:nNnTF
2587     { \@outerparskip } > { \l__scripture_center_belowskip_tl }
2588     { \skip_zero:N \@topsepadd }
2589     {
2590         \skip_set:Nn \@topsepadd
2591         {
2592             \__scripture_skip_diff:NN
2593             \l__scripture_center_belowskip_tl \@outerparskip
2594         }
2595     }
2596     \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
2597     \legacy_if_set_false:n { @nparlist }
2598     \endlist
2599     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2600     \hook_gput_next_code:nn { para / begin }
2601     { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2602 }

```

(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.

```

2603 \cs_new_protected:Nn \_scripture_flushright_begin:n
2604 {
2605   \bool_if:NT \l__scripture_active_inner_bool
2606   {
2607     \msg_error:nn { scripture } { nested-environment }
2608   }
2609   \bool_set_true:N \l__scripture_active_inner_bool
2610   \str_set:Nn \l__scripture_currenvir_str { flushright }
2611   \tl_if_novalue:nF {#1}
2612   {
2613     \keys_set:nn
2614       { scripture / flushright }
2615       {#1}
2616   }
2617   \legacy_if:nTF { @newlist }
2618   {
2619     \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
2620   }
2621   { \dim_zero:N \l__scripture_outer_itemindent_dim }
2622   \cs_set_eq:NN \par \para_end:
2623   \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
2624   \list { }
2625   {
2626     \bool_lazy_or:nnF
2627       { \mode_if_vertical_p: }
2628       { \g__scripture_para_mode_vertical_bool }
2629       { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2630     \dim_zero:N \labelsep
2631     \dim_set:Nn \leftmargin { \l__scripture_flushright_leftmargin_tl }
2632     \dim_set:Nn \rightmargin { \l__scripture_flushright_rightmargin_tl }
2633     \dim_set:Nn \itemindent { - \l__scripture_outer_itemindent_dim }
2634     \dim_zero:N \listparindent
2635     \skip_set_eq:NN \parsep \parskip
2636     \skip_zero:N \partopsep
2637     \dim_compare:nNnTF
2638       { \parskip } > { \l__scripture_flushright_aboveskip_tl }
2639       { \skip_zero:N \topsep }
2640       {
2641         \skip_set:Nn \topsep
2642         {
2643           \_scripture_skip_diff:NN
2644             \l__scripture_flushright_aboveskip_tl \parskip
2645         }
2646       }
2647     \_scripture_setup_list_noindent:n { \c_zero_dim }
2648   }
2649   \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2650   { \addvspace { \baselineskip } }
2651   \int_gzero:N \g__scripture_chap_par_prevgraf_int
2652   \raggedleft
2653   \bool_if:NT \l__scripture_verse_para_bool

```

```

2654     {
2655       \cs_set_eq:NN \par \__scripture_para_par:
2656       \cs_set_protected:Npn \\\
2657         {
2658           \mode_if_horizontal:TF
2659             { \unskip }
2660             { \@nolnerr }
2661         }
2662     }
2663     \item
2664     \relax
2665 }

```

(End of definition for __scripture_flushright_begin:n.)

__scripture_flushright_end: End a flushright section in a scripture quotation.

```

2666 \cs_new_protected:Nn \__scripture_flushright_end:
2667 {
2668   \cs_set_eq:NN \para_end:
2669   \dim_gset:Nn \g__scripture_final_line_dim { \@totalleftmargin + \linewidth }
2670   \dim_compare:nNnTF
2671     { \@outerparskip } > { \l__scripture_flushright_belowskip_tl }
2672     { \skip_zero:N \@topsepadd }
2673     {
2674       \skip_set:Nn \@topsepadd
2675       {
2676         \__scripture_skip_diff:NN
2677         \l__scripture_flushright_belowskip_tl \@outerparskip
2678       }
2679     }
2680   \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
2681   \legacy_if_set_false:n { @nparlist }
2682   \endlist
2683   \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2684   \hook_gput_next_code:nn { para / begin }
2685     { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2686 }

```

(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.

```

2687 \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.

```

2688 \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.

```

2689 \cs_new_protected:Nn \__scripture_hanging_ch:n
2690 {
2691   \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
2692   \bool_if:NTF \l__scripture_chap_show_bool
2693   {
2694     \bool_if:NTF \l__scripture_chap_drop_bool
2695     { \__scripture_hanging_chap:n {#1} }
2696     {
2697       \bool_if:NTF \l__scripture_chap_para_bool
2698       { \__scripture_para_chap:n {#1} }
2699       { \__scripture_nodrop_chap:n {#1} }
2700     }
2701   }
2702   {
2703     \bool_if:NTF \l__scripture_chap_para_bool
2704     { \__scripture_para_chap:n {#1} }
2705     { \__scripture_nodrop_chap:n {#1} }
2706   }
2707 }

```

(End of definition for __scripture_hanging_ch:n.)

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

```

2708 \cs_new_protected:Nn \__scripture_hanging_chap:n
2709 {
2710   \dim_set:Nn \l__scripture_hanging_chap_sep_dim { \l__scripture_chap_sep_tl }
2711   \__scripture_drop_chap_set_up:n {#1}
2712   \dim_compare:nNnTF
2713   { \g__scripture_chap_width_dim + \l__scripture_hanging_chap_sep_dim }
2714   <
2715   { \leftmargin }
2716   {
2717     \bool_if:NT \l__scripture_hanging_chapter_indent_bool
2718     {
2719       \dim_set:Nn \l__scripture_hanging_chap_sep_dim
2720       { \leftmargin - \g__scripture_chap_width_dim }
2721     }
2722   }
2723   {
2724     \parshape 3 ~
2725     \dim_eval:n
2726     {
2727       \@totalleftmargin - \leftmargin + \g__scripture_chap_width_dim +
2728       \l__scripture_hanging_chap_sep_dim
2729     } ~
2730     \dim_eval:n
2731     {
2732       \linewidth + \leftmargin - \g__scripture_chap_width_dim -
2733       \l__scripture_hanging_chap_sep_dim
2734     } ~
2735     \dim_eval:n
2736     {
2737       \@totalleftmargin - \leftmargin + \g__scripture_chap_width_dim +

```

```

2738         \l__scripture_hanging_chap_sep_dim
2739     } ~
2740     \dim_eval:n
2741     {
2742         \linewidth + \leftmargin - \g__scripture_chap_width_dim -
2743         \l__scripture_hanging_chap_sep_dim
2744     } ~
2745     \@totalleftmargin ~ \linewidth
2746 }
2747 \par
2748 \bool_if:NT \l__scripture_chap_show_bool
2749 {
2750     \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
2751     {
2752         \addvspace
2753         {
2754             \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
2755         }
2756     }
2757 }
2758 \bool_if:NTF \l__scripture_hanging_chapter_indent_bool
2759 {
2760     \dim_set:Nn \l__scripture_hanging_chap_sep_dim
2761     {
2762         \l__scripture_hanging_chap_sep_dim +
2763         \l__scripture_hanging_parindent_saved_dim
2764     }
2765     \__scripture_nohang:
2766 }
2767 {
2768     \mode_leave_vertical:
2769     \dim_compare:nNnTF
2770     { \g__scripture_chap_width_dim + \l__scripture_hanging_chap_sep_dim }
2771     >
2772     { \l__scripture_hanging_leftmargin_tl }
2773     {
2774         \dim_compare:nNnTF
2775         {
2776             \g__scripture_chap_width_dim +
2777             \l__scripture_hanging_chap_sep_dim
2778         }
2779         >
2780         { \leftmargin }
2781         {
2782             \skip_horizontal:n { \l__scripture_hanging_hang_tl }
2783         }
2784         {
2785             \skip_horizontal:n
2786             {
2787                 \l__scripture_hanging_chap_sep_dim +
2788                 \g__scripture_chap_width_dim -
2789                 \l__scripture_hanging_leftmargin_tl
2790             }
2791         }
2792     }
2793 }

```

```

2792     }
2793     {
2794         \dim_set:Nn \l__scripture_hanging_chap_sep_dim
2795         {
2796             \l__scripture_hanging_leftmargin_tl -
2797             \g__scripture_chap_width_dim
2798         }
2799     }
2800 }
2801 \hbox_set:Nn \l__scripture_chap_box
2802 {
2803     \box_move_down:nn
2804     { \baselineskip }
2805     {
2806         \hbox_overlap_left:n
2807         {
2808             \box_use:N \l__scripture_chap_tmp_box
2809             \skip_horizontal:N \l__scripture_hanging_chap_sep_dim
2810         }
2811     }
2812 }
2813 \box_set_ht:Nn \l__scripture_chap_box { \c_zero_dim }
2814 \box_set_dp:Nn \l__scripture_chap_box { \c_zero_dim }
2815 \group_begin:
2816 \hook_use:n { scripture / chap / before }
2817 \box_use:N \l__scripture_chap_box
2818 \hook_use:n { scripture / chap / after }
2819 \group_end:
2820 \bool_if:NTF \l__scripture_chap_show_verse_bool
2821 {
2822     \__scripture_verse_output:n { 1 }
2823 }
2824 { \__scripture_insert_verse_mark: }
2825 \__scripture_pilcrow_output:
2826 \bool_if:NT \l__scripture_verse_para_bool
2827 {
2828     \hook_gput_next_code:nn { para / after }
2829     {
2830         \int_gset_eq:NN \g__scripture_chap_par_prevgraf_int \prevgraf
2831         \int_compare:nNnT
2832             { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2833             { \penalty 10000 }
2834         \hook_gput_next_code:nn { para / before }
2835         {
2836             \int_set_eq:NN \clubpenalty \l__scripture_clubpenalty_saved_int
2837         }
2838     }
2839 }
2840 \hook_gput_next_code:nn { para / begin }
2841 {
2842     \int_compare:nNnTF
2843         { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
2844         {
2845             \dim_compare:nNnTF

```

```

2846 { \l__scripture_hanging_leftmargin_tl }
2847 <
2848 {
2849   \g__scripture_chap_width_dim +
2850   \l__scripture_hanging_chap_sep_dim
2851 }
2852 {
2853   \parshape 2 ~
2854   \dim_eval:n
2855   {
2856     \@totalleftmargin - \leftmargin +
2857     \g__scripture_chap_width_dim +
2858     \l__scripture_hanging_chap_sep_dim
2859   } ~
2860   \dim_eval:n
2861   {
2862     \linewidth + \leftmargin - \g__scripture_chap_width_dim -
2863     \l__scripture_hanging_chap_sep_dim
2864   } ~
2865   \@totalleftmargin ~ \linewidth
2866 }
2867 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2868 \hook_gput_next_code:nn { para / begin }
2869 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2870 }
2871 { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
2872 \int_gzero:N \g__scripture_chap_par_prevgraf_int
2873 }
2874 }

```

(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.

```

2875 \cs_new_protected:Nn \__scripture_hanging_versen
2876 {
2877   \mode_if_vertical:TF
2878   {
2879     \tl_set_eq:NN \l_tmpa_tl \l__scripture_versesep_tl
2880     \tl_set_eq:NN
2881     \l__scripture_versesep_tl \l__scripture_hanging_versesep_tl
2882     \bool_lazy_and:nnTF
2883     { \legacy_if_p:n { @newlist } }
2884     { \g__scripture_suppress_next_pilcrow_bool }
2885     {
2886       \strut
2887       \bool_if:NTF \l__scripture_hanging_versesep_right_bool
2888       {
2889         \hbox_overlap_left:n
2890         {
2891           \__scripture_versesep_output:n {#1}
2892         }
2893       }
2894     }

```

```

2895         \hbox_overlap_left:n
2896         {
2897             \hbox_overlap_right:n
2898             {
2899                 \__scripture_vers_output:n {#1}
2900             }
2901             \skip_horizontal:N \l__scripture_hanging_leftmargin_tl
2902         }
2903     }
2904 }
2905 {
2906     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
2907     \strut
2908     \bool_if:NTF \l__scripture_hanging_vers_right_bool
2909     {
2910         \hbox_overlap_left:n
2911         {
2912             \__scripture_vers_output:n {#1}
2913         }
2914     }
2915     {
2916         \hbox_overlap_left:n
2917         {
2918             \hbox_overlap_right:n
2919             {
2920                 \__scripture_vers_output:n {#1}
2921             }
2922             \skip_horizontal:N \l__scripture_hanging_leftmargin_tl
2923         }
2924     }
2925     \__scripture_pilcrow_output:
2926 }
2927 \__scripture_insert_vers_mark:
2928 \tl_set_eq:NN \l__scripture_vers_sep_tl \l_tmpa_tl
2929 }
2930 { \__scripture_vers_output:n {#1} }
2931 }

```

(End of definition for __scripture_hanging_vers:n.)

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

```

2932 \cs_new_protected_nopar:Nn \__scripture_nohang:
2933 {
2934     \legacy_if:nTF { @newlist }
2935     { \skip_horizontal:n { \l__scripture_hanging_hang_tl } }
2936     { \__scripture_noindent_saved: }
2937     \hbox_to_wd:nn { \l__scripture_hanging_parindent_saved_dim } { }
2938     \peek_analysis_map_inline:n
2939     {
2940         \bool_lazy_or:nnT
2941         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
2942         { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
2943         {

```



```

2944         \bool_gset_true:N \g__scripture_suppress_next_verse_para_bool
2945         \__scripture_set_pilcrow_hook:
2946     }
2947     \peek_analysis_map_break:n {##1}
2948 }
2949 }

```

(End of definition for __scripture_nohang:.)

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

```

2950 \cs_new_protected:Nn \__scripture_hanging_begin:n
2951 {
2952     \bool_if:NT \l__scripture_active_inner_bool
2953     {
2954         \msg_error:nn { scripture } { nested-environment }
2955     }
2956     \bool_set_true:N \l__scripture_active_inner_bool
2957     \str_set:Nn \l__scripture_currenvir_str { hanging }
2958     \tl_if_novalue:nF {#1}
2959     {
2960         \keys_set:nn
2961             { scripture / hanging }
2962             {#1}
2963     }
2964     \DeclareDocumentCommand { \ch } { s o m }
2965     {
2966         \tl_if_novalue:nF {##2}
2967         {
2968             \tl_set:Nn \l__scripture_X_char_tl {##2}
2969         }
2970         \__scripture_hanging_ch:n {##3}
2971     }
2972     \bool_if:NTF \l__scripture_verse_para_bool
2973     { \cs_set_eq:NN \vs \__scripture_hanging_verse_para:n }
2974     { \cs_set_eq:NN \vs \__scripture_hanging_verse:n }
2975     \cs_set_eq:NN \nohang \__scripture_nohang:
2976     \dim_set_eq:NN \l__scripture_hanging_parindent_saved_dim \parindent
2977     \legacy_if:nTF { @newlist }
2978     {
2979         \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
2980     }
2981     { \dim_zero:N \l__scripture_outer_itemindent_dim }
2982     \cs_set_eq:NN \par \para_end:
2983     \hook_gclear_next_code:n { para / before }
2984     \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
2985     \list { }
2986     {
2987         \bool_lazy_or:nnF
2988             { \mode_if_vertical_p: }
2989             { \g__scripture_para_mode_vertical_bool }
2990             { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
2991         \dim_zero:N \labelsep
2992         \dim_set:Nn \leftmargin
2993         {

```

```

2994         \l__scripture_hanging_leftmargin_tl +
2995         \l__scripture_hanging_hang_tl
2996     }
2997     \dim_set:Nn \rightmargin { \l__scripture_hanging_rightmargin_tl }
2998     \dim_set:Nn \itemindent
2999     {
3000         - \l__scripture_hanging_hang_tl -
3001         \l__scripture_outer_itemindent_dim
3002     }
3003     \dim_set:Nn \listparindent { - \l__scripture_hanging_hang_tl }
3004     \skip_set_eq:NN \parsep \parskip
3005     \skip_zero:N \partopsep
3006     \dim_compare:nNnTF
3007     { \parskip } > { \l__scripture_hanging_aboveskip_tl }
3008     { \skip_zero:N \topsep }
3009     {
3010         \skip_set:Nn \topsep
3011         {
3012             \__scripture_skip_diff:NN
3013             \l__scripture_hanging_aboveskip_tl \parskip
3014         }
3015     }
3016     \__scripture_setup_list_noindent:n { - \l__scripture_hanging_hang_tl }
3017 }
3018 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3019 { \addvspace { \baselineskip } }
3020 \int_gzero:N \g__scripture_chap_par_prevgraf_int
3021 \item
3022 \relax
3023 }

```

(End of definition for __scripture_hanging_begin:n.)

__scripture_hanging_end: End a hanging section in a scripture quotation.

```

3024 \cs_new_protected:Nn \__scripture_hanging_end:
3025 {
3026     \__scripture_calc_final_line_length:
3027     \dim_compare:nNnTF
3028     { \@outerparskip } > { \l__scripture_hanging_belowskip_tl }
3029     { \skip_zero:N \@topsepadd }
3030     {
3031         \skip_set:Nn \@topsepadd
3032         {
3033             \__scripture_skip_diff:NN
3034             \l__scripture_hanging_belowskip_tl \@outerparskip
3035         }
3036     }
3037     \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
3038     \legacy_if_set_false:n { @nparlist }
3039     \endlist
3040     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3041     \hook_gput_next_code:nn { para / begin }
3042     { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
3043 }

```

(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.

```
3044 \cs_new_protected:Npn \__scripture_vbox_set_top:Nw #1
3045 {
3046   \setbox #1 \vtop
3047   \c_group_begin_token
3048   \color_group_begin:
3049 }
```

(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.

```
3050 \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.

```
3051 \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.

```
3052 \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.

```
3053 \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.

```
3054 \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.

```
3055 \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.

```
3056 \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 `verse/para` and/or `compact` options are in use.

```

3057 \cs_set_eq:Nc \_scripture_mid_para_chap_end_orig:n { end ~ }
3058 \cs_new_protected_nopar:Nn \_scripture_mid_para_chap_end:n
3059 {
3060   \_scripture_mid_para_chap_end_orig:n {#1}
3061   \bool_if:NT \l_scripture_compact_bool
3062     { \c_space_token }
3063   \peek_remove_spaces:n
3064     {
3065       \peek_analysis_map_inline:n
3066       {
3067         \bool_case:n
3068         {
3069           { \exp_args:No \token_if_letter_p:N {##1} }
3070           { \_scripture_pilcrow_output: }
3071           { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
3072           { \_scripture_set_pilcrow_hook: }
3073           { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
3074           { \_scripture_set_pilcrow_hook: }
3075         }
3076         \peek_analysis_map_break:n {##1}
3077       }
3078     }
3079 }

```

(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
\l_scripture_mid_para_chap_split_top_box
3080 \box_new:N \l_scripture_mid_para_chap_box
3081 \box_new:N \l_scripture_mid_para_chap_split_box
3082 \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.

```

3083 \cs_new_protected:Nn \_scripture_mid_para_drop_chap_output:
3084 {
3085   \unskip
3086   \int_set_eq:NN \l_scripture_mid_para_chap_spacefactor_int \spacefactor
3087   \strut
3088   \par
3089   \box_set_to_last:N \l_scripture_mid_para_chap_line_box
3090   \hbox_set:Nn \l_scripture_mid_para_chap_snap_box
3091     {
3092       \hbox_unpack_drop:N \l_scripture_mid_para_chap_line_box
3093       \unskip
3094       \unskip
3095       \unpenalty
3096     }
3097   \nointerlineskip
3098   \dim_compare:nNnTF

```

```

3099 {
3100   \box_wd:N \l__scripture_mid_para_chap_snap_box +
3101   \g__scripture_chap_width_dim +
3102   \l__scripture_chap_sep_tl
3103 }
3104 <
3105 { \linewidth - \leftskip - \rightskip - 3em }
3106 {
3107   \__scripture_drop_chap_output:
3108   \hbox_unpack_drop:N \l__scripture_mid_para_chap_snap_box
3109   \int_set_eq:NN \spacefactor \l__scripture_mid_para_chap_spacefactor_int
3110   \c_space_tl
3111   \__scripture_verse_output:n { 1 }
3112 }
3113 {
3114   \group_begin:
3115   \noindent
3116   \skip_horizontal:n { - \leftskip }
3117   \hbox_unpack_drop:N \l__scripture_mid_para_chap_snap_box
3118   \skip_zero:N \parfillskip
3119   \par
3120   \group_end:
3121   \__scripture_drop_chap_output:
3122   \bool_if:NTF \l__scripture_midparachap_show_verse_bool
3123   {
3124     \__scripture_verse_output:n { 1 }
3125   }
3126   { \__scripture_insert_verse_mark: }
3127 }
3128 }

```

(End of definition for __scripture_mid_para_drop_chap_output:.)

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

```

3129 \cs_new_protected:Nn \__scripture_mid_para_chap_begin:n
3130 {
3131   \str_set:Nn \l__scripture_currenvir_str { midparachap }
3132   \tl_if_no_value:nF {#1}
3133   {
3134     \keys_set:nn
3135     { scripture / midparachap }
3136     {#1}
3137   }
3138   \bool_lazy_and:nnT
3139   { \l__scripture_chap_show_verse_bool }
3140   { \l__scripture_chap_drop_bool }
3141   {
3142     \bool_set_true:N \l__scripture_midparachap_show_verse_bool
3143     \bool_set_false:N \l__scripture_chap_show_verse_bool
3144   }
3145   \par
3146   \bool_set_true:N \l__scripture_mid_para_chap_bool
3147   \__scripture_vbox_set_top:Nw \l__scripture_mid_para_chap_box
3148   \dim_zero:N \leftskip

```

```

3149 \dim_zero:N \rightskip
3150 \dim_set_eq:NN
3151 \l__scripture_mid_para_chap_totalleftmargin_dim \@totalleftmargin
3152 \dim_zero:N \@totalleftmargin
3153 \dim_set_eq:NN \hsize \linewidth
3154 \skip_zero:N \parskip
3155 }

```

(End of definition for __scripture_mid_para_chap_begin:n.)

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

```

3156 \cs_new_protected:Nn \__scripture_mid_para_chap_end:
3157 {
3158   \par
3159   \dim_gset_eq:NN \g__scripture_mid_para_chap_prevdepth_dim \prevdepth
3160   \box_set_to_last:N \l__scripture_mid_para_chap_line_box
3161   \hbox_set:Nn \l__scripture_mid_para_chap_snap_box
3162   {
3163     \hbox_unpack:N \l__scripture_mid_para_chap_line_box
3164     \unskip
3165     \unskip
3166     \unpenalty
3167   }
3168   \dim_gset:Nn \g__scripture_final_line_dim
3169   {
3170     \box_wd:N \l__scripture_mid_para_chap_snap_box +
3171     \l__scripture_mid_para_chap_totalleftmargin_dim
3172   }
3173   \box_clear:N \l__scripture_mid_para_chap_snap_box
3174   \nointerlineskip
3175   \int_compare:nNnT
3176   { \prevgraf } < { 3 }
3177   {
3178     \noindent
3179     \skip_horizontal:n
3180     { \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl }
3181     \dim_gadd:Nn \g__scripture_final_line_dim
3182     { \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl }
3183   }
3184   \box_use_drop:N \l__scripture_mid_para_chap_line_box
3185   \vbox_set_end:

```

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

```

3186 \mode_if_inner:TF
3187 {
3188   \noindent
3189   \box_use_drop:N \l__scripture_mid_para_chap_box
3190 }
3191 {
3192   \dim_compare:nNnTF
3193   { \box_ht_plus_dp:N \l__scripture_mid_para_chap_box }
3194   >
3195   { \pagegoal - \pagetotal - \baselineskip }
3196   {
3197     \int_set:Nn \vbadness { 10000 }

```

```

3198         \vbox_set_split_to_ht:NNn
3199         \l__scripture_mid_para_chap_split_box
3200         \l__scripture_mid_para_chap_box
3201         { \pagegoal - \pagetotal - \baselineskip }
3202     \vbox_set_top:Nn \l__scripture_mid_para_chap_split_top_box
3203     { \vbox_unpack_drop:N \l__scripture_mid_para_chap_split_box }
3204     \mark_get_marks_for_reinsertion:nNN
3205     { \box_use:N \l__scripture_mid_para_chap_split_top_box }
3206     \l_tmpa_tl \l_tmpb_tl
3207     \noindent
3208     \box_use_drop:N \l__scripture_mid_para_chap_split_top_box
3209     \tl_use:N \l_tmpa_tl
3210     \tl_use:N \l_tmpb_tl
3211     \goodbreak
3212     \mark_get_marks_for_reinsertion:nNN
3213     { \box_use:N \l__scripture_mid_para_chap_box }
3214     \l_tmpa_tl \l_tmpb_tl
3215     \box_use_drop:N \l__scripture_mid_para_chap_box
3216     \tl_use:N \l_tmpa_tl
3217     \tl_use:N \l_tmpb_tl
3218 }
3219 {
3220     \noindent
3221     \box_use_drop:N \l__scripture_mid_para_chap_box
3222 }
3223 }
3224 \par
3225 \dim_set_eq:NN \prevdepth \g__scripture_mid_para_chap_prevdepth_dim
3226 \int_zero:N \prevgraf
3227 \skip_vertical:n { - \parskip - \baselineskip }
3228 \noindent
3229 \skip_horizontal:N \g__scripture_final_line_dim
3230 \int_gset_eq:NN
3231     \g__scripture_chap_par_prevgraf_int
3232     \g__scripture_mid_para_chap_prevgraf_int
3233 \int_gset:Nn \g__scripture_current_group_level_a_int
3234     { \currentgrouplevel - \c_one_int }
3235 \__scripture_reset_par_at_group_level:
3236 }

```

(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.

```

3237 \cs_new_protected:Nn \__scripture_narrow_begin:n
3238 {
3239     \bool_if:NT \l__scripture_active_inner_bool
3240     {
3241         \msg_error:nn { scripture } { nested-environment }
3242     }
3243     \bool_set_true:N \l__scripture_active_inner_bool
3244     \str_set:Nn \l__scripture_currenvir_str { narrow }
3245     \tl_if_novalue:nF {#1}

```

```

3246 {
3247   \keys_set:nn
3248   { scripture / narrow }
3249   {#1}
3250 }
3251 \legacy_if:nTF { @newlist }
3252 {
3253   \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
3254 }
3255 { \dim_zero:N \l__scripture_outer_itemindent_dim }
3256 \cs_set_eq:NN \par \para_end:
3257 \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
3258 \list { }
3259 {
3260   \dim_zero:N \labelsep
3261   \dim_set:Nn \leftmargin { \l__scripture_narrow_leftmargin_tl }
3262   \dim_set:Nn \rightmargin { \l__scripture_narrow_rightmargin_tl }
3263   \dim_set_eq:NN \listparindent \parindent
3264   \bool_lazy_or:nnTF
3265   { \mode_if_vertical_p: }
3266   { \g__scripture_para_mode_vertical_bool }
3267   {
3268     \dim_set:Nn \itemindent
3269     { \parindent - \l__scripture_outer_itemindent_dim }
3270     \__scripture_setup_list_noindent:n { \parindent }
3271   }
3272   {
3273     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3274     \dim_set:Nn \itemindent { - \l__scripture_outer_itemindent_dim }
3275     \__scripture_setup_list_noindent:n { \c_zero_dim }
3276   }
3277   \skip_set_eq:NN \parsep \parskip
3278   \skip_zero:N \partopsep
3279   \dim_compare:nNnTF
3280   { \parskip } > { \l__scripture_narrow_aboveskip_tl }
3281   { \skip_zero:N \topsep }
3282   {
3283     \skip_set:Nn \topsep
3284     {
3285       \__scripture_skip_diff:NN
3286       \l__scripture_narrow_aboveskip_tl \parskip
3287     }
3288   }
3289 }
3290 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3291 { \addvspace { \baselineskip } }
3292 \int_gzero:N \g__scripture_chap_par_prevgraf_int
3293 \bool_if:NT \l__scripture_verse_para_bool
3294 { \cs_set_eq:NN \par \__scripture_para_par: }
3295 \item
3296 \relax
3297 }

```

(End of definition for __scripture_narrow_begin:n.)

`\__scripture_narrow_end:` End a narrow section in a scripture quotation.

```

3298 \cs_new_protected:Nn \__scripture_narrow_end:
3299 {
3300   \cs_set_eq:NN \par \para_end:
3301   \__scripture_calc_final_line_length:
3302   \dim_compare:nNnTF
3303     { \@outerparskip } > { \l__scripture_narrow_belowskip_tl }
3304     { \skip_zero:N \@topsepadd }
3305     {
3306       \skip_set:Nn \@topsepadd
3307       {
3308         \__scripture_skip_diff:NN
3309         \l__scripture_narrow_belowskip_tl \@outerparskip
3310       }
3311     }
3312   \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
3313   \legacy_if_set_false:n { @nparlist }
3314   \endlist
3315   \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3316   \hook_gput_next_code:nn { para / begin }
3317     { \parshape 1 ~ \@totalleftmargin ~ \linewidth }
3318 }

```

(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.

```

3319 \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.

```

3320 \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.

```

3321 \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
3322 \dim_new:N \l__scripture_poetry_chap_indent_dim

```

```

3323 \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.

```

3324 \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.

```
3325 \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.

```
3326 \cs_new_protected:Nn \__scripture_poetry_text_right:nn
3327 {
3328   {
3329     \unskip
3330     \hfil
3331     \penalty 50
3332     \skip_horizontal:n {#1}
3333     \hbox:n {}
3334     \nobreak
3335     \hfill
3336     #2
3337     \skip_zero:N \parfillskip
3338     \int_zero:N \finalhyphendemerits
3339   }
3340 }
```

(End of definition for __scripture_poetry_text_right:nn.)

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

```
3341 \cs_new_protected:Nn \__scripture_selah_output:
3342 {
3343   \group_begin:
3344   \l__scripture_selah_font_tl
3345   \__scripture_poetry_text_right:nn
3346   { \l__scripture_selah_sep_tl }
3347   { \__scripture_selah_format:n { \l__scripture_selah_text_tl } }
3348   \group_end:
3349 }
```

(End of definition for __scripture_selah_output:.)

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

```
3350 \cs_new_protected:Nn \__scripture_poetry_ch:n
3351 {
3352   \bool_if:NTF \l__scripture_chap_show_bool
3353   {
3354     \bool_if:NTF \l__scripture_chap_drop_bool
3355     { \__scripture_poetry_chap:n {#1} }
3356     {
3357       \bool_if:NTF \l__scripture_chap_para_bool
3358       { \__scripture_para_chap:n {#1} }
3359       { \__scripture_poetry_nodrop_chap:n {#1} }
3360     }
3361   }
3362   {
3363     \bool_if:NTF \l__scripture_chap_para_bool
3364     { \__scripture_para_chap:n {#1} }
```

```

3365         { \_scripture_poetry_nodrop_chap:n {#1} }
3366     }
3367 }

```

(End of definition for _scripture_poetry_ch:n.)

_scripture_poetry_chap:n Position a drop chapter in a **poetry** environment.

```

3368 \cs_new_protected:Nn \_scripture_poetry_chap:n
3369 {
3370     \_scripture_drop_chap_set_up:n {#1}
3371     \int_gset_eq:NN \g_scripture_current_group_level_c_int \currentgrouplevel
3372     \_scripture_zero_chap_par_prevgraf_at_group_level:
3373     \bool_if:NT \l_scripture_chap_show_bool
3374     {
3375         \skip_if_eq:nnF { \l_scripture_chapter_aboveskip_tl } { \c_zero_skip }
3376         {
3377             \addvspace
3378             {
3379                 \skip_eval:n { \l_scripture_chapter_aboveskip_tl }
3380             }
3381         }
3382     }
3383     \cs_if_eq:NNTF \vs \_scripture_poetry_mode_vertical_versen
3384     {
3385         \dim_zero:N \l_scripture_poetry_chap_indent_dim
3386         \noindent
3387     }
3388     {
3389         \dim_set:Nn \l_scripture_poetry_chap_indent_dim
3390         { \l_scripture_poetry_indent_tl }
3391         \mode_leave_vertical:
3392     }
3393     \dim_set:Nn \l_scripture_poetry_chap_sep_dim { \l_scripture_chap_sep_tl }
3394     \dim_compare:nNnT
3395     { \g_scripture_chap_width_dim + \l_scripture_chap_sep_tl }
3396     <
3397     {
3398         \l_scripture_poetry_leftmargin_tl +
3399         \l_scripture_poetry_chap_indent_dim
3400     }
3401     {
3402         \dim_set:Nn \l_scripture_poetry_chap_sep_dim
3403         {
3404             \l_scripture_poetry_leftmargin_tl +
3405             \l_scripture_poetry_chap_indent_dim -
3406             \g_scripture_chap_width_dim
3407         }
3408     }
3409     \hbox_set:Nn \l_scripture_chap_box
3410     {
3411         \box_move_down:nn
3412         { \baselineskip }
3413         {
3414             \hbox_overlap_left:n

```

```

3415         {
3416         \box_use:N \l__scripture_chap_tmp_box
3417         \skip_horizontal:N \l__scripture_poetry_chap_sep_dim
3418         }
3419     }
3420 }
3421 \box_set_ht:Nn \l__scripture_chap_box { \c_zero_dim }
3422 \box_set_dp:Nn \l__scripture_chap_box { \c_zero_dim }
3423 \dim_set:Nn \l__scripture_poetry_chap_parshape_correction_dim
3424 {
3425     \g__scripture_chap_width_dim +
3426     \l__scripture_poetry_chap_sep_dim -
3427     \l__scripture_poetry_leftmargin_tl -
3428     \l__scripture_poetry_chap_indent_dim
3429 }
3430 \parshape 2 ~
3431 \dim_eval:n
3432 {
3433     \@totalleftmargin + \l__scripture_poetry_chap_parshape_correction_dim
3434 } ~
3435 \dim_eval:n
3436 { \linewidth - \l__scripture_poetry_chap_parshape_correction_dim } ~
3437 \dim_eval:n
3438 { \@totalleftmargin + \l__scripture_poetry_big_indent_tl } ~
3439 \dim_eval:n
3440 { \linewidth - \l__scripture_poetry_big_indent_tl }

```

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

```

3441 \dim_compare:nNnT
3442 { \g__scripture_chap_width_dim + \l__scripture_poetry_chap_sep_dim }
3443 >
3444 { \l__scripture_poetry_leftmargin_tl }
3445 {
3446     \hook_gput_next_code:nn { scripture / poetry / para / after }
3447     {
3448         \int_compare:nNnT
3449         { \l__scripture_poetry_prevgraf_int } = { \c_one_int }
3450         {
3451             \cs_if_eq:NNTF \vs \__scripture_poetry_mode_vertical_verses:n

```

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

```

3452     {
3453         \dim_compare:nNnTF
3454         {
3455             \g__scripture_chap_width_dim +
3456             \l__scripture_chap_sep_tl
3457         }
3458         <
3459         { \l__scripture_poetry_leftmargin_tl }
3460         {
3461             \dim_zero:N
3462             \l__scripture_poetry_chap_parshape_correction_dim
3463         }
3464         {
3465             \dim_set:Nn

```

```

3466         \l__scripture_poetry_chap_parshape_correction_dim
3467     {
3468         \g__scripture_chap_width_dim +
3469         \l__scripture_chap_sep_tl -
3470         \l__scripture_poetry_leftmargin_tl
3471     }
3472 }
3473 \int_compare:nNnT
3474 { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3475 {
3476     \parshape 2 ~
3477     \dim_eval:n
3478     {
3479         \@totalleftmargin +
3480         \l__scripture_poetry_chap_parshape_correction_dim
3481     } ~
3482     \dim_eval:n
3483     {
3484         \linewidth -
3485         \l__scripture_poetry_chap_parshape_correction_dim
3486     } ~
3487     \dim_eval:n
3488     {
3489         \@totalleftmargin +
3490         \l__scripture_poetry_big_indent_tl
3491     } ~
3492     \dim_eval:n
3493     {
3494         \linewidth -
3495         \l__scripture_poetry_big_indent_tl
3496     }
3497 }
3498 }

```

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

```

3499 {
3500     \dim_compare:nNnTF
3501     {
3502         \g__scripture_chap_width_dim + \l__scripture_chap_sep_tl
3503     }
3504     <
3505     {
3506         \l__scripture_poetry_leftmargin_tl +
3507         \l__scripture_poetry_indent_tl
3508     }
3509     {
3510         \dim_zero:N
3511         \l__scripture_poetry_chap_parshape_correction_dim
3512     }
3513     {
3514         \dim_set:Nn
3515         \l__scripture_poetry_chap_parshape_correction_dim
3516         {
3517             \g__scripture_chap_width_dim +
3518             \l__scripture_chap_sep_tl -

```

```

3519         \l__scripture_poetry_leftmargin_tl -
3520         \l__scripture_poetry_indent_tl
3521     }
3522 }
3523 \parshape 2 ~
3524 \dim_eval:n
3525 {
3526     \@totalleftmargin +
3527     \l__scripture_poetry_chap_parshape_correction_dim
3528 } ~
3529 \dim_eval:n
3530 {
3531     \linewidth -
3532     \l__scripture_poetry_chap_parshape_correction_dim
3533 } ~
3534 \dim_eval:n
3535 {
3536     \@totalleftmargin + \l__scripture_poetry_big_indent_tl
3537 } ~
3538 \dim_eval:n
3539 { \linewidth - \l__scripture_poetry_big_indent_tl }
3540 }
3541 }
3542 }
3543 }
3544 \group_begin:
3545 \hook_use:n { scripture / chap / before }
3546 \box_use:N \l__scripture_chap_box
3547 \hook_use:n { scripture / chap / after }
3548 \group_end:
3549 \bool_lazy_or:nnTF
3550 { \l__scripture_chap_show_verse_bool }
3551 { \l__scripture_poetry_midparachap_show_verse_bool }
3552 { \__scripture_verse_output:n { 1 } }
3553 { \__scripture_insert_verse_mark: }
3554 \hook_use:n { scripture / poetry / pilcrow }
3555 }

```

(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.

```

3556 \cs_new_protected:Nn \__scripture_poetry_nodrop_chap:n
3557 {
3558     \tl_gset:Nn \scripturecurrentchapter {#1}
3559     \tl_gset:Nn \scripturecurrentverse { 1 }
3560     \group_begin:
3561     \bool_if:NT \l__scripture_chap_show_bool
3562     {
3563         \skip_if_eq:nnF { \l__scripture_chapter_aboveskip_tl } { \c_zero_skip }
3564         {
3565             \addvspace
3566             {
3567                 \skip_eval:n { \l__scripture_chapter_aboveskip_tl }
3568             }

```

```

3569     }
3570   }
3571   \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical_verse:n
3572   { \noindent }
3573   \bool_if:NTF \l__scripture_chap_show_bool
3574   {
3575     \hbox_overlap_left:n
3576     {
3577       \hook_use:n { scripture / chap / before }
3578       \group_begin:
3579       \l__scripture_chap_font_tl
3580       \str_if_eq:VnF \l__scripture_chapter_colour_tl { . }
3581       { \exp_args:NV \color_select:n \l__scripture_chapter_colour_tl }
3582       \__scripture_select_language:n { chapter }
3583       \__scripture_chap_format:n {#1}
3584       \group_end:
3585       \skip_horizontal:N \l__scripture_chap_sep_tl
3586       \hook_use:n { scripture / chap / after }
3587       \bool_lazy_or:nnTF
3588       { \l__scripture_chap_show_verse_bool }
3589       { \l__scripture_poetry_midparachap_show_verse_bool }
3590       { \__scripture_verse_output:n { 1 } }
3591       { \__scripture_insert_verse_mark: }
3592     }
3593   }
3594   {
3595     \hook_use:n { scripture / chap / before }
3596     \hook_use:n { scripture / chap / after }
3597     \vs { 1 }
3598   }
3599   \group_end:
3600   \hook_use:n { scripture / poetry / pilcrow }
3601 }

```

(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.

```

3602 \cs_new_protected:Nn \__scripture_poetry_mode_vertical_verse:n
3603 {
3604   \group_begin:
3605   \tl_set_eq:NN \l__scripture_verse_sep_tl \l__scripture_poetry_verse_sep_tl
3606   \hook_gclear_next_code:n { scripture / pilcrow }
3607   \strut
3608   \int_compare:nNnTF { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3609   {
3610     \__scripture_verse_output:n {#1}
3611   }
3612   {
3613     \bool_if:NTF \l__scripture_poetry_verse_right_bool
3614     {
3615       \hbox_overlap_left:n
3616       {
3617         \__scripture_verse_output:n {#1}
3618       }

```

```

3619     }
3620     {
3621         \hbox_overlap_left:n
3622         {
3623             \hbox_overlap_right:n
3624             {
3625                 \__scripture_verse_output:n {#1}
3626             }
3627             \skip_horizontal:N \l__scripture_poetry_leftmargin_tl
3628         }
3629     }
3630     \__scripture_insert_verse_mark:
3631 }
3632 \group_end:
3633 \bool_if:NF \g__scripture_suppress_next_pilcrow_bool
3634 { \hook_use:n { scripture / poetry / pilcrow } }
3635 \cs_set_eq:NN \vs \__scripture_verse_output:n
3636 }

```

(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.

```

3637 \cs_new_protected:Nn \__scripture_poetry_mode_horizontal_verse:n
3638 {
3639     \group_begin:
3640     \tl_set_eq:NN \l__scripture_verse_sep_tl \l__scripture_poetry_verse_sep_tl
3641     \int_compare:nNnTF { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3642     {
3643         \__scripture_poetry_mode_vertical_verse:n {#1}
3644     }
3645     {
3646         \skip_horizontal:n { - \listparindent }
3647         \__scripture_poetry_mode_vertical_verse:n {#1}
3648         \skip_horizontal:N \listparindent
3649     }
3650     \group_end:
3651     \cs_set_eq:NN \vs \__scripture_verse_output:n
3652 }

```

(End of definition for __scripture_poetry_mode_horizontal_verse:n.)

__scripture_poetry_par: par macro for poetry sections

```

3653 \cs_new_protected:Nn \__scripture_poetry_par:
3654 {
3655     \para_end:
3656     \int_set:Nn \l__scripture_poetry_prevgraf_int { \int_use:N \prevgraf }
3657     \parshape 2 ~
3658     \@totalleftmargin ~
3659     \linewidth ~
3660     \dim_eval:n
3661     { \@totalleftmargin + \l__scripture_poetry_big_indent_tl } ~
3662     \dim_eval:n
3663     { \linewidth - \l__scripture_poetry_big_indent_tl }
3664     \hook_use:n { scripture / poetry / para / after }
3665 }

```


(End of definition for _scripture_poetry_par:.)

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

```

3666 \group_begin:
3667   \char_set_catcode_active:N \^M
3668   \cs_gset_protected_nopar:Nn \_scripture_obeylines:
3669     {
3670       \char_set_catcode_active:N \^M
3671       \cs_set_eq:NN \^M \_scripture_obeylines_eol:
3672     }
3673 \group_end:
3674 \cs_new_protected:Nn \_scripture_obeylines_eol:
3675 {
3676   \bool_lazy_and:nnTF
3677   { \mode_if_vertical_p: }
3678   {
3679     \bool_not_p:n { \l__scripture_poetry_secondhalf_bool }
3680   }
3681   {
3682     \cs_set_eq:NN \vs \_scripture_poetry_mode_vertical_verse:n
3683     \hook_gput_next_code:nn { scripture / poetry / para / after }
3684     { \bool_set_false:N \l__scripture_poetry_midparachap_show_verse_bool }
3685   }
3686   { \cs_set_eq:NN \vs \_scripture_poetry_mode_horizontal_verse:n }
3687   \bool_set_false:N \l__scripture_poetry_secondhalf_bool
3688   \peek_analysis_map_inline:n
3689   {
3690     \bool_case:nF
3691     {
3692       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \ch }
3693       {
3694         \bool_set_eq:NN
3695         \l__scripture_poetry_midparachap_show_verse_bool
3696         \l__scripture_midparachap_show_verse_bool
3697         \_scripture_poetry_par:
3698         \peek_analysis_map_break:n {##1}
3699       }
3700       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newch }
3701       {
3702         \bool_set_eq:NN
3703         \l__scripture_poetry_midparachap_show_verse_bool
3704         \l__scripture_midparachap_show_verse_bool
3705         \_scripture_poetry_par:
3706         \peek_analysis_map_break:n {##1}
3707       }
3708       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \end }
3709       { \peek_analysis_map_break:n {##1} }
3710       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \extraskip }
3711       { \peek_analysis_map_break:n {##1} }
3712       { \exp_args:No \token_if_eq_meaning_p:NN {##1} \nopilcrow }
3713       {
3714         \_scripture_poetry_par:
3715         \cs_if_eq:NNT \vs \_scripture_poetry_mode_vertical_verse:n
3716         { \noindent }

```

```

3717         \peek_analysis_map_break:n
3718         {
3719             \__scripture_fix_protrusion:o {##1}
3720         }
3721     }
3722 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \vs }
3723 {
3724     \__scripture_poetry_par:
3725     \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical Verse:n
3726     { \noindent }
3727     \peek_analysis_map_break:n {##1}
3728 }
3729 { \exp_args:No \token_if_eq_meaning_p:NN {##1} \newvs }
3730 {
3731     \__scripture_poetry_par:
3732     \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical Verse:n
3733     { \noindent }
3734     \peek_analysis_map_break:n {##1}
3735 }
3736 {
3737     \exp_args:No
3738     \token_if_eq_meaning_p:NN {##1} \__scripture_obeylines_eol:
3739 }
3740 { \cs_set_eq:NN \vs \__scripture_poetry_mode_vertical Verse:n }
3741 }
3742 {
3743     \__scripture_poetry_par:
3744     \cs_if_eq:NNT \vs \__scripture_poetry_mode_vertical Verse:n
3745     { \noindent }
3746     \cs_set_eq:NN \vs \__scripture_verse_output:n
3747     \bool_if:NT \l__scripture_verse_para_bool
3748     {
3749         \__scripture_pilcrow_output:
3750         \bool_set_false:N \l__scripture_verse_para_bool
3751         \bool_set_false:N \l__scripture_pilcrow_bool
3752     }
3753     \peek_analysis_map_break:n
3754     {
3755         \__scripture_fix_protrusion:o {##1}
3756     }
3757 }
3758 }
3759 }

```

(End of definition for __scripture_obeylines:.)

__scripture_poetry_begin: Begin a poetry section in a scripture quotation.

```

3760 \cs_new_protected:Nn \__scripture_poetry_begin:
3761 {
3762     \bool_if:NT \l__scripture_active_inner_bool
3763     {
3764         \msg_error:nn { scripture } { nested-environment }
3765     }
3766     \bool_set_true:N \l__scripture_active_inner_bool

```

```

3767 \str_set:Nn \l__scripture_currenvir_str { poetry }
3768 \bool_gset_false:N \g__scripture_poetry_midparachap_show_verse_bool
3769 \DeclareDocumentCommand { \ch } { s o m }
3770 {
3771   \tl_if_novalue:nF {##2}
3772   {
3773     \tl_set:Nn \l__scripture_X_char_tl {##2}
3774   }
3775   \__scripture_poetry_ch:n {##3}
3776 }
3777 \cs_set_eq:NN \vs \__scripture_poetry_mode_vertical_verse:n
3778 \cs_set_eq:NN \selah \__scripture_selah_output:
3779 \cs_set_protected_nopar:Npn \textright ##1
3780 {
3781   \__scripture_poetry_text_right:nn
3782   { \l__scripture_text_right_sep_tl } {##1}
3783 }
3784 \legacy_if:nTF { @newlist }
3785 {
3786   \dim_add:Nn \l__scripture_outer_itemindent_dim { \itemindent }
3787 }
3788 { \dim_zero:N \l__scripture_outer_itemindent_dim }
3789 \cs_set_eq:NN \par \para_end:
3790 \cs_set_eq:NN \noindent \__scripture_noindent_saved:
3791 \hook_gc_clear_next_code:n { para / before }
3792 \hook_gput_next_code:nn { scripture / poetry / pilcrow }
3793 {
3794   \__scripture_pilcrow_output:
3795   \bool_set_false:N \l__scripture_pilcrow_bool
3796   \bool_set_false:N \l__scripture_verse_para_bool
3797 }
3798 \int_gset_eq:NN \g__scripture_current_group_level_b_int \currentgrouplevel
3799 \list { }
3800 {
3801   \bool_lazy_or:nnF
3802   { \mode_if_vertical_p: }
3803   { \g__scripture_para_mode_vertical_bool }
3804   { \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool }
3805   \dim_zero:N \labelsep
3806   \dim_set:Nn \leftmargin { \l__scripture_poetry_leftmargin_tl }
3807   \dim_set:Nn \rightmargin { \l__scripture_poetry_rightmargin_tl }
3808   \dim_set:Nn \itemindent
3809   {
3810     \l__scripture_poetry_indent_tl - \l__scripture_outer_itemindent_dim
3811   }
3812   \dim_set:Nn \listparindent { \l__scripture_poetry_indent_tl }
3813   \skip_zero:N \parsep
3814   \skip_zero:N \partopsep
3815   \dim_compare:nNnTF { \parskip } > { \l__scripture_poetry_aboveskip_tl }
3816   { \skip_zero:N \topsep }
3817   {
3818     \skip_set:Nn \topsep
3819     {
3820       \__scripture_skip_diff:NN

```

```

3821         \l__scripture_poetry_aboveskip_tl \parskip
3822     }
3823 }
3824 \__scripture_setup_list_noindent:n { \l__scripture_poetry_indent_tl }
3825 }
3826 \int_compare:nNnT { \g__scripture_chap_par_prevgraf_int } = { \c_one_int }
3827 { \addvspace { \baselineskip } }
3828 \int_gzero:N \g__scripture_chap_par_prevgraf_int
3829 \raggedright
3830 \dim_set:Nn \parindent { \l__scripture_poetry_indent_tl }
3831 \item
3832 \relax
3833 }

```

(End of definition for __scripture_poetry_begin:.)

__scripture_poetry_end: End a poetry section in a scripture quotation.

```

3834 \cs_new_protected:Nn \__scripture_poetry_end:
3835 {
3836     \__scripture_calc_final_line_length:
3837     \dim_compare:nNnTF
3838     { \@outerparskip } > { \l__scripture_poetry_belowskip_tl }
3839     { \skip_zero:N \@topsepadd }
3840     {
3841         \skip_set:Nn \@topsepadd
3842         {
3843             \__scripture_skip_diff:NN
3844             \l__scripture_poetry_belowskip_tl \@outerparskip
3845         }
3846     }
3847     \skip_gset_eq:NN \g__scripture_prev_inner_below_skip \@topsepadd
3848     \legacy_if_set_false:n { @nparlist }
3849     \endlist
3850     \bool_gset_true:N \g__scripture_suppress_next_pilcrow_bool
3851     \hook_gc_clear_next_code:n { scripture / poetry / pilcrow }
3852     \bool_gset_true:N \g__scripture_poetry_end_bool
3853     \hook_gput_next_code:nn { para / begin }
3854     {
3855         \bool_gset_false:N \g__scripture_poetry_end_bool
3856         \parshape 1 ~ \@totalleftmargin ~ \linewidth
3857     }
3858 }

```

(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, **\scripturecurrentchapter** for example. They are initialised from the **book**, **chapter** and **verse** options to the main **scripture** environment. **\scripturecurrentverse**

```

3859 \tl_new:N \scripturecurrentbook
3860 \tl_new:N \scripturecurrentchapter
3861 \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.

```

3862 \NewDocumentCommand { \scripturesetup } { m }
3863 {
3864   \keys_set:nn { scripture } {#1}
3865 }

```

(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.

```

3866 \NewDocumentCommand { \scripturestyle } { s m m }
3867 {
3868   \__scripture_setup_style:nnn {#1} {#2} {#3}
3869 }

```

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

`\textscripture` #1 : `<reference>` of quotation.
#2 : `<options>` to set.
#3 : `<quotation>` text.
Set an inline quotation. This is equivalent to calling the **`scripture`** environment with the **`inline`** option set.

```

3870 \NewDocumentCommand { \textscripture } { o o +m }
3871 {
3872   \tl_if_novalue:nTF {#2}
3873   {
3874     \begin { scripture } [ #1 ] [ inline ]
3875   }
3876   {
3877     \begin { scripture } [ #1 ] [ inline, #2 ]
3878   }
3879   #3
3880   \end { scripture }
3881 }

```

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

`scripture` #1 : `<reference>` of quotation.
#2 : `<options>` to set.
Format a scripture quotation with optional reference.

```

3882 \NewDocumentEnvironment { scripture } { o o }
3883 {
3884   \__scripture_begin:nn {#1} {#2}

```

```

#1 : <options> to set.
center 3885 \DeclareDocumentEnvironment { center } { o }
        3886 {
        3887     \__scripture_center_begin:n {##1}
        3888     \__scripture_reference_start_peek:
        3889     }
        3890     { \__scripture_center_end: }

#1 : <options> to set.
flushright 3891 \DeclareDocumentEnvironment { flushright } { o }
            3892 {
            3893     \__scripture_flushright_begin:n {##1}
            3894     \__scripture_reference_start_peek:
            3895     }
            3896     { \__scripture_flushright_end: }

#1 : <options> to set.
hanging 3897 \DeclareDocumentEnvironment { hanging } { o }
         3898 {
         3899     \__scripture_hanging_begin:n {##1}
         3900     \__scripture_reference_start_peek:
         3901     }
         3902     { \__scripture_hanging_end: }

midparachap 3903 \DeclareDocumentEnvironment { midparachap } { o }
            3904 {
            3905     \__scripture_mid_para_chap_begin:n {##1}
            3906     \__scripture_reference_start_peek:
            3907     }
            3908     { \__scripture_mid_para_chap_end: }

#1 : <options> to set.
narrow 3909 \DeclareDocumentEnvironment { narrow } { o }
       3910 {
       3911     \__scripture_narrow_begin:n {##1}
       3912     \__scripture_reference_start_peek:
       3913     }
       3914     { \__scripture_narrow_end: }

#1 : <options> to set.
poetry 3915 \DeclareDocumentEnvironment { poetryaux } { o }
        3916 {
        3917     \tl_if_novalue:nF {##1}
        3918     {
        3919         \keys_set:nn
        3920         { scripture / poetry }
        3921         {##1}
        3922     }
        3923     \__scripture_poetry_begin:
        3924     \__scripture_reference_start_peek:
        3925     }

```

The poetryaux environment is required because __scripture_obeylines: breaks the reading of optional arguments.

```

3926     { \__scripture_poetry_end: }
3927 \DeclareDocumentEnvironment { poetry } { }
3928 {
3929     \__scripture_obeylines:
3930     \poetryaux
3931 }
3932 { \endpoetryaux }

3933 \bool_if:NT \l__scripture_inline_bool
3934 {
3935     \tl_set:Nn \l__scripture_aboveskip_tl { Opt }
3936     \tl_set:Nn \l__scripture_belowskip_tl { Opt }
3937     \tl_set:Nn \l__scripture_leftmargin_tl { Opt }
3938     \tl_set:Nn \l__scripture_rightmargin_tl { Opt }
3939     \cs_set_eq:NN \__scripture_ref_format:n \__scripture_inline_ref_format:n
3940     \tl_set_eq:NN \l__scripture_ref_sep_tl \l__scripture_inline_ref_sep_tl
3941     \cs_set_eq:NN
3942         \__scripture_version_format:n \__scripture_inline_version_format:n
3943     \tl_set_eq:NN
3944         \l__scripture_version_delim_tl \l__scripture_inline_version_delim_tl
3945     \bool_set_false:N \l__scripture_chap_para_bool
3946     \bool_set_true:N \l__scripture_compact_bool
3947     \bool_set_false:N \l__scripture_heading_show_bool
3948 }
3949 \bool_if:NT \l__scripture_chap_para_bool
3950 {
3951     \RenewDocumentEnvironment { midparachap } { o }
3952     { \ignorespaces }
3953     { \unskip }
3954 }
3955 \bool_if:NT \l__scripture_verse_para_bool
3956 {
3957     \cs_set_eq:NN \vs \__scripture_verse_para_output:n
3958     \cs_set_eq:NN \par \__scripture_para_par:
3959     \cs_set_eq:NN \@par \__scripture_para_par:
3960     \cs_set_eq:NN \@doendpe \__scripture_doendpe:
3961     \cs_set_eq:NN \noindent \__scripture_para_noindent:
3962     \cs_set_protected:Npn \\\
3963     {
3964         \mode_if_horizontal:TF
3965         { \unskip }
3966         { \@nolnerr }
3967     }
3968     \RenewDocumentEnvironment { midparachap } { o }
3969     {
3970         \str_set:Nn \l__scripture_currenvir_str { midparachap }
3971         \cs_set_eq:NN \noindent \__scripture_noindent_saved:
3972         \cs_set_eq:cN { end ~ } \__scripture_mid_para_chap_end:n
3973         \__scripture_reference_start_peek:
3974     }
3975     {
3976         \para_end:
3977         \bool_gset_false:N \g__scripture_suppress_next_pilcrow_bool
3978     }

```

```

3979     }
3980 \bool_if:NT \l__scripture_compact_bool
3981 {
3982     \cs_set_eq:NN \noindent \__scripture_para_noindent:
3983     \bool_if:NF \l__scripture_verse_para_bool
3984     {
3985         \bool_set_false:N \l__scripture_chap_drop_bool
3986         \bool_set_false:N \l__scripture_chap_drop_llap_bool
3987         \bool_set_false:N \l__scripture_chap_drop_smash_bool
3988         \bool_set_false:N \l__scripture_chap_para_bool
3989         \cs_set_eq:NN \vs \__scripture_verse_compact_output:n
3990     }
3991     \bool_gset:Nn \g__scripture_mode_vertical_inner_bool
3992     { \mode_if_vertical_p: }
3993     \RenewDocumentEnvironment { center } { o }
3994     {
3995         \str_set:Nn \l__scripture_currenvir_str { center }
3996         \__scripture_reference_start_peek:
3997     }
3998     { \__scripture_compact_end_inner: }
3999     \RenewDocumentEnvironment { flushright } { o }
4000     {
4001         \str_set:Nn \l__scripture_currenvir_str { flushright }
4002         \__scripture_reference_start_peek:
4003     }
4004     { \__scripture_compact_end_inner: }
4005     \RenewDocumentEnvironment { hanging } { o }
4006     {
4007         \str_set:Nn \l__scripture_currenvir_str { hanging }
4008         \__scripture_reference_start_peek:
4009     }
4010     { \__scripture_compact_end_inner: }
4011     \RenewDocumentEnvironment { midparachap } { o }
4012     {
4013         \str_set:Nn \l__scripture_currenvir_str { midparachap }
4014         \cs_set_eq:cN { end ~ } \__scripture_mid_para_chap_end:n
4015         \__scripture_reference_start_peek:
4016     }
4017     { \__scripture_compact_end_inner: }
4018     \RenewDocumentEnvironment { narrow } { o }
4019     {
4020         \str_set:Nn \l__scripture_currenvir_str { narrow }
4021         \__scripture_reference_start_peek:
4022     }
4023     { \__scripture_compact_end_inner: }
4024     \RenewDocumentEnvironment { poetry } { o }
4025     {
4026         \str_set:Nn \l__scripture_currenvir_str { poetry }
4027         \__scripture_reference_start_peek:
4028     }
4029     { \__scripture_compact_end_inner: }
4030     \cs_set_eq:NN \ \ \unskip
4031     \cs_set_eq:NN \par \__scripture_compact_par:
4032     \cs_set_eq:NN \extraskip \relax

```



```

4033     \cs_set_eq:NN \nohang \relax
4034     \cs_set_protected:Npn \selah
4035     {
4036         \group_begin:
4037         \l__scripture_selah_font_tl
4038         \c_space_tl
4039         \__scripture_selah_format:n { \l__scripture_selah_text_tl }
4040         \c_space_tl
4041         \group_end:
4042     }
4043     \cs_set_protected_nopar:Npn \textright ##1
4044     {
4045         \unskip
4046         \c_space_tl
4047         ##1
4048     }
4049 }
4050 \__scripture_reference_start_peek:
4051 }
4052 {
4053     \bool_lazy_or:nnT
4054     { \l__scripture_compact_bool } { \l__scripture_verse_para_bool }
4055     { \cs_set_eq:NN \par \para_end: }
4056     \__scripture_end:
4057 }

```

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

```

4058 %</package>

```

Change History

v1.0		Add verse/firstsep option	27
General: First public release	1	Add verse/first option	26
v1.1		Add feature to highlight translator	
General: Add		added words	4
\scripturecurrentchapter and		Bug fixes	1
\scripturecurrentverse		Remove reference/delim option	13
variables	8	v2.0	
Add \selah function	8	General: Add \scripturestyle	
Add chapter/format option	15	function	8
Add chapter/hide option	15	Add inline/reference/format	
Add chapter/showverse option	17	option	20
Add chapter/show option	17	Add inline/reference/sep option	21
Add compact option	18	Add inline/version/delim option	21
Add poetry/verse/left option	33	Add inline/version/format	
Add poetry/verse/right option	34	option	21
Add poetry/verse/sep option	34	Add style option	26
Add verse/hide option	27	Bug fixes	1
Add verse/sep option with small		Switch scripture to a L ^A T _E X list	
default	28	based environment	1
Add verse/show option	28	v2.1	
Add before and after hooks for		General: Add \heading function	6
verses and chapters	12	Add \nopilcrow function	7
Bug fixes	1	Add after option	14
v1.2		Add before option	14
General: Add \textright function	8	Add chapter/align option	15
Add chapter/colour option	15	Add chapter/para/aboveskip	
Add colour option	18	option	16
Add language/variant option	22	Add chapter/para/belowskip	
Add language option	21	option	16
Add name/font option	22	Add chapter/para/indent option	16
Add name/format option	22	Add chapter/para option	16
Add redletter/colour option	24	Add chapter/smash option	17
Add redletter option	24	Add chapter/valign option	17
Add reference/colour option	24	Add chapter/xchar option	18
Add textright/sep option	26	Add defaults option	19
Add verse/colour option	26	Add hanging/chapter/indent	
Add divine name feature	6	option	30
Add red letter feature	7	Add hanging/verse/left option	31
Bug fixes	1	Add hanging/verse/right option	31
v1.3		Add hanging/verse/sep option	31
General: Add \nofirstverse function	7	Add heading/aboveskip option	19
Add \textscripture function	9	Add heading/afterindent option	19
Add added/font option	13	Add heading/align option	19
Add added/format option	14	Add heading/belowskip option	19
Add inline/begin option	20	Add heading/font option	19
Add inline/end option	20	Add heading/format option	19
Add inline option	20	Add heading/hide option	20
Add reference/align option	24	Add heading/show option	20
Add reference/newline option	24	Add pilcrow/sep option	23
Add verse/firstformat option	27	Add pilcrow option	23

Add reference/position option . .	24	Add chapter/language/variant option	16
Add reference/start/inline option	25	Add chapter/language option . . .	15
Add reference/start/newline option	25	Add inline/font option	20
Add reference/start/sep option .	25	Add reference/language/variant option	24
Add reference/start/vsep option	25	Add reference/language option . .	24
Add verse/para option	27	Add verse/language/variant option	27
Add version/position option . . .	29	Add verse/language option	27
Add feature to add pilcrows at the start of paragraphs.	23	Add verse/presep option	28
Add feature to insert simple headings	6	Add version/language/variant option	29
Add feature to place the reference at the start of the quotation. . . .	24	Add version/language option . . .	29
Add feature to set chapters in a paragraph of their own	16	v2.5 General: Add \canonicalheading function	4
Add feature to set each verse on a new line.	27	Add canonicalheading/afterindent option	14
Bug fixes	1	Add canonicalheading/align option	14
v2.2 General: Add book option	14	Add canonicalheading/belowskip option	14
Add chapter option	14	Add canonicalheading/font option	14
Add verse option	26	Add canonicalheading/format option	14
Add auto chapter numbering	5	Add canonicalheading/aboveskip option	14
Add auto verse numbering	9	Add chapter/presep option	16
Bug fixes	1	Add poetry/secondhalf option . .	33
Support footnotes	1	Add feature to insert canonical headings	4
v2.3 General: Add mark/verse/format option	22		
Add marks for headings and verses	13		
v2.4 General: Add chapter/aboveskip option	15		

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, 2571, 2656, 3962, 4030
`\^` 3667, 3670

A

`\abovedisplayskip` 2002, 2003
`\abovedisplayskip` 2003
`aboveskip` (option) *13*
`\added` *4*, 2267
`added/font` (option) *13*
`added/format` (option) *14*
`\addvspace` 845, 849,
996, 1494, 1590, 1677, 1720, 1725,
2053, 2077, 2101, 2126, 2565, 2650,
2752, 3019, 3291, 3377, 3565, 3827
`after` (option) *14*

B

`\baselineskip` 843,
845, 915, 920, 1451, 1751, 1790,
2002, 2029, 2565, 2650, 2804, 3019,
3195, 3201, 3227, 3291, 3412, 3827
`before` (option) *14*
`\begin` .. 1043, 1138, 1167, 1237, 3874, 3877
`\belowdisplayskip` 2004
`\belowdisplayskip` 2005
`belowskip` (option) *14*
`\bfseries` *15*, *24*, 133, 356, 613, 660
`\bigskipamount` .. *16*, *19*, 161, 228, 618, 630
`book` (option) *14*
`bool` commands:
`\bool_case:n` 1235, 1297, 3067
`\bool_case:nTF` 1016, 1151, 3690
`\bool_gset:Nn` 3991
`\bool_gset_eq:NN`
.... 957, 960, 1678, 1683, 1728, 1733
`\bool_gset_false:N`
1229, 1323, 1375, 1936, 1967, 1981,
1982, 1989, 2201, 2222, 2228, 2229,
2230, 2458, 2691, 3768, 3855, 3977
`\bool_gset_true:N`
. 794, 958, 1238, 1264, 1279, 1291,
1333, 1358, 1491, 1650, 1679, 1730,
1738, 1959, 2227, 2251, 2264, 2544,
2599, 2629, 2683, 2906, 2944, 2990,
3040, 3273, 3315, 3804, 3850, 3852
`\bool_if:N` 733, 786,
808, 833, 976, 1011, 1022, 1028,

1034, 1045, 1066, 1083, 1100, 1111,
1122, 1125, 1127, 1169, 1209, 1211,
1276, 1334, 1344, 1356, 1372, 1475,
1541, 1546, 1558, 1565, 1603, 1627,
1661, 1762, 1765, 1873, 1879, 1885,
1935, 1953, 1983, 1988, 2061, 2070,
2090, 2110, 2119, 2153, 2155, 2179,
2185, 2193, 2233, 2246, 2262, 2269,
2290, 2305, 2332, 2336, 2388, 2520,
2568, 2605, 2653, 2692, 2694, 2697,
2703, 2717, 2748, 2758, 2820, 2826,
2887, 2908, 2952, 2972, 3061, 3122,
3239, 3293, 3352, 3354, 3357, 3363,
3373, 3561, 3573, 3613, 3633, 3747,
3762, 3933, 3949, 3955, 3980, 3983
`\bool_if:nTF` 2252
`\bool_if_p:N` 2473
`\bool_lazy_all:nTF` 979, 1892
`\bool_lazy_and:nnTF` .. 1003, 2158,
2357, 2463, 2472, 2882, 3138, 3676
`\bool_lazy_any:nTF` 1699, 2339
`\bool_lazy_or:nnTF` 790, 853,
969, 1007, 1049, 1173, 1261, 1272,
1985, 2393, 2433, 2541, 2626, 2940,
2987, 3264, 3549, 3587, 3801, 4053
`\bool_new:N` 772,
773, 774, 775, 1287, 1353, 1852,
1853, 1854, 1970, 3050, 3320, 3321
`\bool_not_p:n`
.... 983, 987, 1005, 1897, 2161, 3679
`\bool_set:Nn` 168
`\bool_set_eq:NN` 3694, 3702
`\bool_set_false:N` 1859,
1864, 1875, 1876, 2094, 2234, 3143,
3684, 3687, 3750, 3751, 3795, 3796,
3945, 3947, 3985, 3986, 3987, 3988
`\bool_set_true:N` 1012,
1860, 2149, 2197, 2295, 2524, 2609,
2956, 3142, 3146, 3243, 3766, 3946
`\c_false_bool` 737
`\g_tmpa_bool`
.... 957, 960, 1678, 1683, 1729, 1734

box commands:

`\box_clear:N` 3173
`\box_gclear:N` 1314, 1330, 2231
`\box_gset_to_last:N` 1328
`\box_ht:N` 1443
`\box_ht_plus_dp:N` 3193

<code>\box_if_empty:N</code>	1881	<code>chapter/xchar (option)</code>	18
<code>\box_move_down:nn</code>	1750, 2803, 3411	char commands:	
<code>\box_new:N</code>	880, 1288, 1429, 1430, 3051, 3055, 3080, 3081, 3082	<code>\char_set_catcode_active:N</code>	3667, 3670
<code>\box_set_dp:Nn</code>	1761, 2814, 3422	<code>\clubpenalty</code>	1319, 1473, 1474, 1833, 2836
<code>\box_set_ht:Nn</code>	1760, 2813, 3421	coffin commands:	
<code>\box_set_to_last:N</code>	3089, 3160	<code>\coffin_new:N</code>	1437
<code>\box_set_wd:Nn</code>	1336	<code>\coffin_typeset:Nnnnn</code>	1609, 1618
<code>\box_use:N</code>	938, 1538, 1755, 2808, 2817, 3205, 3213, 3416, 3546	color (option)	18
<code>\box_use_drop:N</code>	1882, 3184, 3189, 3208, 3215, 3221	color commands:	
<code>\box_wd:N</code>	909, 928, 1472, 3100, 3170	<code>\color_group_begin:</code>	3048
<code>\break</code>	858	<code>\color_select:n</code>	886, 1359, 1365, 1374, 1381, 1467, 1571, 1690, 1890, 2304, 3581
C			
<code>\canonicalheading</code>	4, 1190, 1649, 2271, 2278	<code>\color_set:nn</code>	2302
<code>canonicalheading/aboveskip (option)</code>	14	colour (option)	18
<code>canonicalheading/afterindent (option)</code>	14	compact (option)	18
<code>canonicalheading/align (option)</code>	14	cs commands:	
<code>canonicalheading/belowskip (option)</code>	14	<code>\cs_generate_variant:Nn</code>	746, 1352
<code>canonicalheading/font (option)</code>	14	<code>\cs_gset_nopar:Nn</code>	2013
<code>canonicalheading/format (option)</code>	14	<code>\cs_gset_protected_nopar:Nn</code>	3668
<code>center (environment)</code>	10, 118	<code>\cs_if_eq:NNTF</code>	1499, 1726, 3383, 3451, 3571, 3715, 3725, 3732, 3744
<code>center/aboveskip (option)</code>	29	<code>\cs_if_exist_use:N</code>	218, 1347, 1349
<code>center/belowskip (option)</code>	29	<code>\cs_new_protected:Nn</code>	727, 731, 753, 818, 829, 851, 881, 901, 940, 953, 999, 1227, 1254, 1289, 1312, 1342, 1354, 1370, 1386, 1439, 1445, 1489, 1553, 1655, 1745, 1778, 1819, 1839, 1855, 1921, 1927, 1949, 1971, 1979, 2000, 2151, 2190, 2379, 2454, 2518, 2581, 2603, 2666, 2689, 2708, 2875, 2950, 3024, 3083, 3129, 3156, 3237, 3298, 3326, 3341, 3350, 3368, 3556, 3602, 3637, 3653, 3674, 3760, 3834
<code>center/leftmargin (option)</code>	29	<code>\cs_new_protected:Npn</code>	3044
<code>center/rightmargin (option)</code>	29	<code>\cs_new_protected_nopar:Nn</code>	779, 2034, 2041, 2086, 2140, 2147, 2932, 3058
<code>\centering</code>	80, 252, 2567	<code>\cs_set_eq:NN</code>	768, 811, 812, 814, 1069, 1076, 1086, 1093, 1103, 1105, 1114, 1116, 2063, 2066, 2112, 2115, 2192, 2223, 2224, 2225, 2243, 2267, 2268, 2278, 2279, 2282, 2289, 2296, 2374, 2387, 2451, 2466, 2511, 2537, 2570, 2583, 2622, 2655, 2668, 2973, 2974, 2975, 2982, 3057, 3256, 3294, 3300, 3635, 3651, 3671, 3682, 3686, 3740, 3746, 3777, 3778, 3789, 3790, 3939, 3941, 3957, 3958, 3959, 3960, 3961, 3971, 3972, 3982, 3989, 4014, 4030, 4031, 4032, 4033, 4055
<code>\ch</code>	5, 50, 1064, 1069, 1070, 1076, 1193, 2235, 2254, 2257, 2964, 3692, 3769	<code>\cs_set_nopar:Npn</code>	769, 770
<code>\ch*</code>	5	<code>\cs_set_protected:Npn</code>	1316, 2226, 2271, 2571, 2656, 3962, 4034
<code>chapter (option)</code>	14		
<code>chapter/aboveskip (option)</code>	15		
<code>chapter/align (option)</code>	15		
<code>chapter/color (option)</code>	15		
<code>chapter/colour (option)</code>	15		
<code>chapter/drop (option)</code>	15		
<code>chapter/font (option)</code>	15		
<code>chapter/format (option)</code>	15		
<code>chapter/hide (option)</code>	15		
<code>chapter/language (option)</code>	15		
<code>chapter/language/variant (option)</code>	16		
<code>chapter/nodrop (option)</code>	16		
<code>chapter/para (option)</code>	16		
<code>chapter/para/aboveskip (option)</code>	16		
<code>chapter/para/belowskip (option)</code>	16		
<code>chapter/para/indent (option)</code>	16		
<code>chapter/presep (option)</code>	16		
<code>chapter/sep (option)</code>	17		
<code>chapter/show (option)</code>	17		
<code>chapter/showverse (option)</code>	17		
<code>chapter/smash (option)</code>	17		
<code>chapter/valign (option)</code>	17		

<code>\cs_set_protected_nopar:Npn</code>	698, 701, 702, 708, 721, 824, 826,
. . . 783, 2280, 2281, 2283, 3779, 4043	1597, 1760, 1761, 2008, 2027, 2562,
<code>\currentgrouplevel</code> . 1480, 1484, 1548,	2647, 2813, 2814, 3275, 3421, 3422
1772, 1783, 1824, 1844, 2513, 2538,	<code>\dimexpr</code> 822
2623, 2984, 3234, 3257, 3371, 3798	
	E
D	<code>\emph</code> 14, 39, 598
<code>\DeclareDocumentCommand</code>	<code>\end</code> 100, 3708, 3880
. 2235, 2244, 2260, 2964, 3769	<code>\endlist</code> 2448, 2598, 2682, 3039, 3314, 3849
<code>\DeclareDocumentEnvironment</code> . . 3885,	<code>\endpoetryaux</code> 3932
3891, 3897, 3903, 3909, 3915, 3927	environments:
defaults (option) 19	<code>center</code> 10, 118
dim commands:	<code>flushright</code> 10, 118
<code>\dim_add:Nn</code>	<code>hanging</code> 10, 118
. . 1670, 2534, 2619, 2979, 3253, 3786	<code>midparachap</code> 11, 118
<code>\dim_compare:nNnTF</code> 840,	<code>narrow</code> 11, 118
907, 926, 1597, 1790, 1792, 2008,	<code>poetry</code> 11, 118
2022, 2027, 2322, 2417, 2552, 2586,	<code>scripture</code> 12, 117
2637, 2670, 2712, 2769, 2774, 2845,	<code>\everydisplay</code> 82, 2217
3006, 3027, 3098, 3192, 3279, 3302,	<code>\everypar</code> 1320, 1326, 1331
3394, 3441, 3453, 3500, 3815, 3837	exp commands:
<code>\dim_eval:n</code> . 822, 1510, 1516, 1522,	<code>\exp_args:NNo</code> 1608, 1617
1528, 2300, 2484, 2490, 2725, 2730,	<code>\exp_args:NnV</code> 2301
2735, 2740, 2854, 2860, 3431, 3435,	<code>\exp_args:No</code> 791,
3437, 3439, 3477, 3482, 3487, 3492,	792, 1018, 1020, 1026, 1032, 1043,
3524, 3529, 3534, 3538, 3660, 3662	1064, 1081, 1098, 1109, 1138, 1153,
<code>\dim_gadd:Nn</code> 3181	1155, 1157, 1159, 1167, 1189, 1193,
<code>\dim_gset:Nn</code>	1195, 1197, 1202, 1237, 1241, 1243,
. . 1471, 2010, 2024, 2584, 2669, 3168	1245, 1247, 1262, 1263, 1273, 1274,
<code>\dim_gset_eq:NN</code> 3159	1299, 1301, 1303, 1305, 1647, 2941,
<code>\dim_gsub:Nn</code> 2416	2942, 3069, 3071, 3073, 3692, 3700,
<code>\dim_max:nn</code> 824, 826	3708, 3710, 3712, 3722, 3729, 3737
<code>\dim_new:N</code> 778, 1431, 1432,	<code>\exp_args:NV</code> 886, 1359,
1433, 1435, 1436, 1438, 1999, 2687,	1365, 1467, 1571, 1690, 1890, 3581
2688, 3052, 3053, 3322, 3323, 3324	<code>\exp_not:n</code> 413
<code>\dim_ratio:nn</code> 1455	extraskip (option) 19
<code>\dim_set:Nn</code> 1230,	<code>\extraskip</code> 5, 2223, 3710, 4032
1292, 1442, 1450, 1452, 1585, 1666,	
1672, 1933, 2015, 2313, 2314, 2315,	F
2319, 2546, 2547, 2548, 2631, 2632,	<code>\finalhyphendemerits</code> 875, 3338
2633, 2710, 2719, 2760, 2794, 2992,	<code>\FirstMark</code> 13
2997, 2998, 3003, 3261, 3262, 3268,	<code>flushright</code> (environment) 10, 118
3274, 3389, 3393, 3402, 3423, 3465,	<code>flushright/aboveskip</code> (option) 30
3514, 3806, 3807, 3808, 3812, 3830	<code>flushright/belowskip</code> (option) 30
<code>\dim_set_eq:NN</code> 1579, 1663,	<code>flushright/leftmargin</code> (option) 30
2309, 2976, 3150, 3153, 3225, 3263	<code>flushright/rightmargin</code> (option) 30
<code>\dim_sub:Nn</code> 1674	<code>font</code> (option) 19
<code>\dim_use:N</code> 2015	<code>\fontsize</code> 1462
<code>\dim_zero:N</code> 1576, 2310,	<code>\footnote</code> 72, 73
2536, 2545, 2549, 2621, 2630, 2634,	<code>\footnotemark</code> 3
2981, 2991, 3148, 3149, 3152, 3255,	<code>\footnotesize</code> 14, 92, 607
3260, 3385, 3461, 3510, 3788, 3805	<code>\footnotetext</code> 3
<code>\c_zero_dim</code> 309, 396, 496, 499,	
508, 512, 531, 578, 648, 669, 697,	G
	<code>\glueexpr</code> 841

<code>\int_gset_eq:NN</code>	1479, 1548, 1772, 1826, 2173, 2468, 2538, 2623, 2830, 2984, 3230, 3257, 3371, 3798
<code>\int_gzero:N</code>	847, 1812, 1847, 2166, 2167, 2216, 2566, 2651, 2872, 3020, 3292, 3828
<code>\int_new:N</code>	776, 1434, 1775, 1776, 1777, 3054, 3056, 3325
<code>\int_set:Nn</code>	. . 2016, 2218, 2220, 2414, 3197, 3656
<code>\int_set_eq:NN</code>	1473, 1832, 2219, 2221, 2836, 3086, 3109
<code>\int_use:N</code>	2016, 2018, 2211, 2213, 2254, 2257, 2265, 3656
<code>\int_zero:N</code>	875, 3226, 3338
<code>\c_one_int</code>	106, 419, 610, 675, 838, 919, 1788, 1828, 2476, 2513, 2564, 2649, 2832, 2843, 3018, 3234, 3290, 3449, 3474, 3608, 3641, 3826
<code>\c_zero_int</code>	2426
<code>\item</code>	2375, 2578, 2663, 3021, 3295, 3831
<code>\itemindent</code>	50, 2309, 2315, 2534, 2548, 2619, 2633, 2979, 2998, 3253, 3268, 3274, 3786, 3808
<code>\itshape</code>	19, 25, 263, 402, 634, 671
K	
<code>\kern</code>	1574, 1906, 1911, 1992, 2510
keys commands:	
<code>\l_keys_choice_str</code>	 65, 115, 194, 237, 346, 374, 484
<code>\keys_define:nn</code>	29, 742
<code>\keys_precompile:nnN</code>	594
<code>\keys_set:nn</code>	2207, 2528, 2613, 2960, 3134, 3247, 3864, 3919
L	
<code>\labelsep</code>	2545, 2630, 2991, 3260, 3805
<code>language</code> (option)	21
<code>language/variant</code> (option)	22
<code>\LastMark</code>	13
<code>leftmargin</code> (option)	22
<code>\leftmargin</code>	962, 1671, 1674, 2313, 2546, 2631, 2715, 2720, 2727, 2732, 2737, 2742, 2780, 2856, 2862, 2992, 3261, 3806
<code>\leftprotrusion</code>	62, 1347, 1349
<code>\leftskip</code>	3105, 3116, 3148
legacy commands:	
<code>\legacy_if:nTF</code>	781, 1256, 1501, 1560, 1785, 1858, 1938, 1955, 2047, 2095, 2308, 2355, 2532, 2617, 2934, 2977, 3251, 3784
<code>\legacy_if_p:n</code>	2883
<code>\legacy_if_set:nn</code>	1706, 2068, 2117
<code>\legacy_if_set_false:n</code>	1322, 1332, 2447, 2597, 2681, 3038, 3313, 3848
<code>\legacy_if_set_true:n</code>	1315, 1710
<code>\linewidth</code>	24, 367, 913, 932, 1518, 1530, 1535, 1641, 1663, 1814, 1943, 1961, 2025, 2045, 2092, 2446, 2483, 2492, 2496, 2585, 2601, 2669, 2685, 2732, 2742, 2745, 2862, 2865, 2867, 2869, 2871, 3042, 3105, 3153, 3317, 3436, 3440, 3484, 3494, 3531, 3539, 3659, 3663, 3856
<code>\list</code>	2311, 2539, 2624, 2985, 3258, 3799
<code>\listparindent</code>	2319, 2549, 2634, 3003, 3263, 3646, 3648, 3812
<code>\LORD</code>	6, 50, 769, 2280
M	
mark commands:	
<code>\mark_get_marks_for_reinsertion:nnN</code>	3204, 3212
<code>\mark_insert:nn</code>	757, 2107, 2129
<code>\mark_new_class:n</code>	751, 752
<code>mark/verse/format</code> (option)	22
marks:	
<code>scripture/heading</code>	13, 49
<code>scripture/verse</code>	13, 49
<code>\medskipamount</code>	14, 16, 19, 29, 30, 32, 33, 55, 89, 165, 222, 260, 490, 493, 502, 505, 515, 518, 551, 554, 563, 566, 603, 606, 619, 628, 633, 695, 696, 699, 700, 703, 704, 712, 713, 716, 717
<code>midparachap</code> (environment)	11, 118
<code>midparachap/hideverse</code> (option)	32
<code>midparachap/showverse</code> (option)	32
mode commands:	
<code>\mode_if_horizontal:TF</code>	 2573, 2658, 3964
<code>\mode_if_inner:TF</code>	3186
<code>\mode_if_vertical:TF</code>	831, 1199, 1204, 1562, 1595, 1676, 1861, 1923, 1929, 1940, 1951, 1957, 2877
<code>\mode_if_vertical_p:</code>	2542, 2627, 2988, 3265, 3677, 3802, 3992
<code>\mode_leave_vertical:</code>	 1605, 1862, 2768, 3391
msg commands:	
<code>\msg_error:nn</code>	 2195, 2522, 2607, 2954, 3241, 3764
<code>\msg_error:nnn</code>	410, 1414, 2080, 2134
<code>\msg_new:nnnn</code>	6, 10, 14, 18, 24
<code>\msg_warning:nnn</code>	1421
N	
<code>\name</code>	6, 50, 768, 2279, 2280, 2281

name/font (option)	22	chapter/color	15
name/format (option)	22	chapter/colour	15
narrow (environment)	11, 118	chapter/drop	15
narrow/aboveskip (option)	32	chapter/font	15
narrow/belowskip (option)	32	chapter/format	15
narrow/leftmargin (option)	32	chapter/hide	15
narrow/rightmargin (option)	33	chapter/language	15
\NeedsTeXFormat	3	chapter/language/variant	16
\newch	5, 14, 1081, 1086, 1087, 1093, 1195, 2244, 3700	chapter/nodrop	16
\newch*	5	chapter/para	16
\NewDocumentCommand ...	3862, 3866, 3870	chapter/para/aboveskip	16
\NewDocumentEnvironment	3882	chapter/para/belowskip	16
\newvs	9, 26, 792, 1109, 1114, 1115, 1116, 1202, 1247, 1263, 1274, 1305, 2260, 2942, 3073, 3729	chapter/para/indent	16
\nobreak	871, 994, 1624, 1698, 1913, 1993, 2075, 2124, 3334	chapter/presep	16
\nofirstverse	7, 2282	chapter/sep	17
\nohang	7, 1026, 1157, 2975, 4033	chapter/show	17
noindent (option)	23	chapter/showverse	17
\noindent	50, 60, 783, 811, 812, 814, 959, 1020, 1155, 1499, 1508, 1682, 1731, 2063, 2066, 2112, 2115, 2192, 2289, 2366, 2387, 2432, 2451, 3115, 3178, 3188, 3207, 3220, 3228, 3386, 3572, 3716, 3726, 3733, 3745, 3790, 3961, 3971, 3982	chapter/smash	17
\nointerlineskip	3097, 3174	chapter/valign	17
\nopilcrow	7, 1032, 1159, 2226, 3712	chapter/xchar	18
O		color	18
\obeylines	3	colour	18
options:		compact	18
aboveskip	13	defaults	19
added/font	13	extraskip	19
added/format	14	flushright/aboveskip	30
after	14	flushright/belowskip	30
before	14	flushright/leftmargin	30
belowskip	14	flushright/rightmargin	30
book	14	font	19
canonicalheading/aboveskip	14	hanging/aboveskip	30
canonicalheading/afterindent	14	hanging/belowskip	30
canonicalheading/align	14	hanging/chapter/indent	30
canonicalheading/belowskip	14	hanging/hang	31
canonicalheading/font	14	hanging/leftmargin	31
canonicalheading/format	14	hanging/rightmargin	31
center/aboveskip	29	hanging/verse/left	31
center/belowskip	29	hanging/verse/right	31
center/leftmargin	29	hanging/verse/sep	31
center/rightmargin	29	heading/aboveskip	19
chapter	14	heading/afterindent	19
chapter/aboveskip	15	heading/align	19
chapter/align	15	heading/belowskip	19
		heading/font	19
		heading/format	19
		heading/hide	20
		heading/show	20
		indent	20
		inline	20
		inline/begin	20
		inline/end	20
		inline/font	20
		inline/reference/format	20
		inline/reference/sep	21

2638, 2644, 3004, 3007, 3013, 3154,	
3227, 3277, 3280, 3286, 3815, 3821	
\partopsep	2320, 2551, 2636, 3005, 3278, 3814
\peek	62
peek commands:	
\peek_analysis_map_break:n	802, 1037,
	1144, 1161, 1219, 1250, 1266, 1283,
	1308, 1651, 2947, 3076, 3698, 3706,
	3709, 3711, 3717, 3727, 3734, 3753
\peek_analysis_map_inline:n	
	788, 1014, 1149, 1233, 1259,
	1270, 1295, 1645, 2938, 3065, 3688
\peek_meaning:NTF	1718
\peek_remove_spaces:n	
	1001, 1231, 1293, 1643, 3063
\penalty	868, 1829, 2833, 3331
pilcrow (option)	23
pilcrow/sep (option)	23
poetry (environment)	11, <u>118</u>
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/secondhalf (option)	33
poetry/verse/left (option)	33
poetry/verse/right (option)	34
poetry/verse/sep (option)	34
\poetryaux	3930
\postdisplaypenalty	2219, 2414
\predisplaypenalty	2218
\preplaysize	2008, 2011, 2015
\prevdepth	99, 3159, 3225
\prevgraf	73, 99, 1826, 2016,
	2174, 2426, 2830, 3176, 3226, 3656
\ProcessKeyOptions	592
\ProvidesExplPackage	4

R

<code>\raggedleft</code>	75, 247, 2652
<code>\raggedright</code>	70, 242, 3829
<code>redletter</code> (option)	24
<code>redletter/color</code> (option)	24
<code>redletter/colour</code> (option)	24
<code>\redletteroff</code>	7, 2225
<code>\redletteron</code>	7, 2224
<code>reference/align</code> (option)	24
<code>reference/color</code> (option)	24
<code>reference/colour</code> (option)	24
<code>reference/font</code> (option)	24
<code>reference/format</code> (option)	24
<code>reference/language</code> (option)	24

reference/language/variant (option) ..	24
reference/newline (option)	24
reference/position (option)	24
reference/sep (option)	25
reference/start/inline (option)	25
reference/start/newline (option)	25
reference/start/sep (option)	25
reference/start/vsep (option)	25
\relax	9, 18, 20, 2376, 2511, 2579, 2664, 3022, 3296, 3832, 4032, 4033
\renewcommand*	8
\RenewDocumentCommand	1070, 1087, 1104, 1115
\RenewDocumentEnvironment	3951, 3968, 3993, 3999, 4005, 4011, 4018, 4024
rightmargin (option)	25
\rightmargin	1671, 2314, 2547, 2632, 2997, 3262, 3807
\rightskip	3105, 3149

S

```

scripture (environment) ..... 12, 117
scripture commands:
  \added ..... 4, 13, 14
  \canonicalheading ..... 4, 14, 123
  \ch ..... 8, 15–18, 24, 32, 51, 105, 116
  \extraskip ..... 18, 19
  \GOD ..... 6
  \heading ..... 3, 6, 19, 20, 35, 49, 122
  \LORD ..... 6
  \name ..... 6, 22
  \nofirstverse ..... 26, 122
  \nopilcrow ..... 23, 122
  \redletteroff ..... 7, 24
  \redletteron ..... 7, 24
  \scripturecurrentchapter . 8, 49, 122
  \scripturecurrentverse ... 8, 49, 122
  \scripturesetup .... 13, 29, 30, 32, 33
  \scripturestyle ..... 26, 122
  \selah ..... 25, 26, 122
  \textright ..... 26, 122
  \textscripture ..... 9, 20, 122
  \vs ..... 7, 8, 17,
    24, 27, 28, 31, 32, 34, 49, 61, 95, 116

```

```

scripture internal commands:
  \l_scripture_aboveskip_tl      ....
                                31, 2322, 2328, 3935
  \l_scripture_active_bool      ....
                                772, 2193, 2197
  \l_scripture_active_inner_bool
                                .... 772, 2520, 2524, 2605, 2609,
                                2952, 2956, 3239, 3243, 3762, 3766
  \_scripture_added:n           2034, 2034, 2267
  \l_scripture_added_font_tl    34, 2037

```

_scripture_added_format:n	37, 2038	_scripture_chap_format:n	
\l_scripture_after_tl	 40, 2452		141, 1469, 1573, 1693, 3583
\l_scripture_before_tl	... 43, 2214	\l_scripture_chap_height_dim	..
_scripture_begin:nn	<u>2190</u> , 2190, 3884		<u>1432</u> , 1450, 1456
\l_scripture_belowskip_tl		\g_scripture_chap_par_prevgraf_-	
.....	46, 2417, 2423, 3936	int	 73, 838, 847,
\g_scripture_book_tl	. 49, 2204, 2209		919, <u>1434</u> , 1788, 1812, 1826, 1828,
_scripture_calc_final_line_-			1847, 2017, 2018, 2166, 2216, 2470,
length:	<u>2000</u> ,		2564, 2566, 2649, 2651, 2830, 2832,
	2000, 2415, 2467, 3026, 3301, 3836		2843, 2872, 3018, 3020, 3231, 3290,
_scripture_canonicalheading:n			3292, 3474, 3608, 3641, 3826, 3828
.....	<u>2041</u> , 2041, 2278	_scripture_chap_par_save_-	
\l_scripture_canonicalheading_-		prevgraf_at_group_level:	
aboveskip_tl	 53, 2053		<u>1486</u> , 1549, <u>1819</u> , 1819, 1836
\l_scripture_canonicalheading_-		\l_scripture_chap_para_aboveskip_-	
afterindent_bool	.. 57, 2069, 2070	tl	 159, 1677
\l_scripture_canonicalheading_-		\l_scripture_chap_para_afterindent_-	
align_tl	... 63, 64, 69, 74, 79, 2056	bool	 169, 1707
\l_scripture_canonicalheading_-		\l_scripture_chap_para_belowskip_-	
belowskip_tl	 87, 2077	tl	 <u>163</u> , 1720, 1725
\l_scripture_canonicalheading_-		\l_scripture_chap_para_bool	...
font_tl	 90, 2057		155, 2179, 2185, 2233, 2697,
_scripture_canonicalheading_-			2703, 3357, 3363, 3945, 3949, 3988
format:n	 94, 2058	\l_scripture_chap_presep_tl	...
\l_scripture_center_aboveskip_-			174, 1597, 1599
tl	 488, 2553, 2559	\l_scripture_chap_sep_tl	
_scripture_center_begin:n			177, 1514, 1520, 1526, 1532,
.....	<u>2518</u> , 2518, 3887		1574, 1756, 1795, 1806, 2488, 2494,
\l_scripture_center_belowskip_-			2502, 2710, 3102, 3180, 3182, 3393,
tl	 491, 2587, 2593		3395, 3456, 3469, 3502, 3518, 3585
_scripture_center_end:		\l_scripture_chap_show_bool	...
.....	<u>2581</u> , 2581, 3890		144, 180, 1565, 1661, 2153,
\l_scripture_center_leftmargin_-			2692, 2748, 3352, 3373, 3561, 3573
tl	 494, 2546	\l_scripture_chap_show_verse_-	
\l_scripture_center_rightmargin_-		bool	 183, 1541,
tl	 497, 2547		1627, 2820, 3139, 3143, 3550, 3588
_scripture_ch_output:nn		\l_scripture_chap_smash_bool	..
.....	<u>2151</u> , 2151, 2241		186, 1603
_scripture_chap:n	. <u>1745</u> , 1745, 2157	\l_scripture_chap_tmp_box	
\l_scripture_chap_box	 <u>1429</u> ,		<u>1430</u> , 1441,
	1538, 1748, 1760, 1761, 2801, 2813,		1443, 1459, 1472, 1755, 2808, 3416
	2814, 2817, 3409, 3421, 3422, 3546	l_scripture_chap_tmp_box	 <u>1429</u>
\l_scripture_chap_drop_bool	...	\g_scripture_chap_width_dim	...
.....	138, 153, 2155,		<u>1433</u> , 1471, 1513, 1519, 1525,
	2234, 2464, 2694, 3140, 3354, 3985		1531, 1794, 1805, 2487, 2493, 2501,
\l_scripture_chap_drop_llap_-			2713, 2720, 2727, 2732, 2737, 2742,
bool	 3986		2770, 2776, 2788, 2797, 2849, 2857,
\l_scripture_chap_drop_smash_-			2862, 3101, 3180, 3182, 3395, 3406,
bool	 3987		3425, 3442, 3455, 3468, 3502, 3517
\l_scripture_chap_font_tl		\l_scripture_chap_X_height_dim	..
.....	131, 1461, 1572, 1691, 3579	..	<u>1435</u> , 1442, 1451, 1457, 1581, 1586
\l_scripture_chap_fontsize_dim		\l_scripture_chapter_aboveskip_-	
.....	<u>1431</u> , 1452, 1463, 1464	tl	. 107, 1492, 1496, 1588, 1592,
			2750, 2754, 3375, 3379, 3563, 3567

<code>\l__scripture_chapter_align_tl .</code>	<code>__scripture_end: . .</code>
..... 113, 114, 119, 124, 1610, 1619	2379, 2379, 4056
<code>\l__scripture_chapter_colour_tl</code>	<code>__scripture_extra_skip:</code>
..... 134, 1466, 1467,	829, 829, 2223
1570, 1571, 1689, 1690, 3580, 3581	<code>\l__scripture_extraskip_tl</code>
<code>\g__scripture_chapter_int</code>	220, 841, 849
... 104, 2202, 2211, 2248, 2254, 2257	<code>\g__scripture_final_line_dim</code>
<code>\l__scripture_chapter_language_-</code>	910, 929, 935, 1999,
<code>tl</code>	2010, 2022, 2024, 2027, 2416, 2500,
146	2507, 2584, 2669, 3168, 3181, 3229
<code>\l__scripture_chapter_language_-</code>	<code>\g__scripture_first_verse_set_-</code>
<code>variant_tl</code>	bool 774, 2201, 2246, 2251, 2262, 2264
150	<code>__scripture_fix_protrusion:n</code>
<code>\l__scripture_chapter_valign_tl</code>	1040, 1145, 1164,
..... 192, 193,	1221, 1342, 1342, 1352, 3719, 3755
198, 203, 208, 1577, 1583, 1611, 1620	<code>\l__scripture_flushright_-</code>
<code>\l__scripture_clubpenalty_saved_-</code>	aboveskip_tl 500, 2638, 2644
<code>int</code>	<code>__scripture_flushright_begin:n</code>
776, 1473, 1833, 2836	2603, 2603, 3893
<code>\l__scripture_colour_tl 97, 2302, 2303</code>	<code>\l__scripture_flushright_-</code>
<code>\l__scripture_compact_bool</code>	belowskip_tl 503, 2671, 2677
..... 101, 976, 1011,	<code>__scripture_flushright_end:</code>
1125, 1209, 1344, 1986, 2161, 2269,	2666, 2666, 3896
2336, 2473, 3061, 3946, 3980, 4054	<code>\l__scripture_flushright_-</code>
<code>__scripture_compact_end_inner:</code>	leftmargin_tl 506, 2631
..... 2454, 2454,	<code>\l__scripture_flushright_-</code>
3998, 4004, 4010, 4017, 4023, 4029	rightmargin_tl 510, 2632
<code>__scripture_compact_par:</code>	<code>\l__scripture_font_tl</code>
..... 1227, 1227, 4031	223, 2292
<code>\g__scripture_current_group_-</code>	<code>__scripture_format_full_ref:nn</code>
<code>level_a_int</code>	881,
1772, 1775, 1783, 2512, 3233	881, 905, 946, 949, 965, 968, 2402, 2407
<code>\g__scripture_current_group_-</code>	<code>__scripture_get_X_height:</code>
<code>level_b_int</code>	1439, 1439, 1449, 1567
..... 1480, 1483, 1548, 1776,	<code>\l__scripture_hanging_aboveskip_-</code>
1824, 2538, 2623, 2984, 3257, 3798	tl 513, 3007, 3013
<code>\g__scripture_current_group_-</code>	<code>__scripture_hanging_begin:n</code>
<code>level_c_int</code>	2950, 2950, 3899
1775, 1844, 3371	<code>\l__scripture_hanging_belowskip_-</code>
<code>\l__scripture_currenvir_str</code>	tl 516, 3028, 3034
..... 799, 810,	<code>__scripture_hanging_ch:n</code>
854, 855, 961, 970, 971, 984, 988,	2689, 2689, 2970
1008, 1009, 1052, 1057, 1132, 1176,	<code>__scripture_hanging_chap:n</code>
1181, 1215, 1239, 1346, 1360, 1363,	2695, 2708, 2708
1376, 1379, 1477, 1632, 1664, 1680,	<code>\l__scripture_hanging_chap_sep_-</code>
1701, 1702, 1703, 1737, 1769, 1899,	dim 2688,
2032, 2043, 2083, 2088, 2137, 2200,	2710, 2713, 2719, 2728, 2733, 2738,
2343, 2347, 2351, 2360, 2480, 2525,	2743, 2760, 2762, 2770, 2777,
2610, 2957, 3131, 3244, 3767, 3970,	2787, 2794, 2809, 2850, 2858, 2863
3995, 4001, 4007, 4013, 4020, 4026	<code>\l__scripture_hanging_chapter_-</code>
<code>\l__scripture_default_options_tl</code>	indent_bool 520, 2717, 2758
..... 593, 726, 729	<code>__scripture_hanging_end:</code>
<code>__scripture_doendpe: 1312, 1312, 3960</code>	3024, 3024, 3902
<code>__scripture_dopeend:</code>	<code>\l__scripture_hanging_hang_tl</code>
61	523, 2782, 2935, 2995, 3000, 3003, 3016
<code>__scripture_drop_chap_output:</code>	
..... 1489, 1489, 1768, 3107, 3121	
<code>__scripture_drop_chap_set_up:n</code>	
..... 1445, 1445, 1747, 2711, 3370	

\l__scripture_hanging_leftmargin_- tl 526, 2772, 2789, 2796, 2846, 2901, 2922, 2994	\l__scripture_language_variant_- tl 1390, 1403, 1409
\l__scripture_hanging_parindent_- saved_dim .. 2687, 2763, 2937, 2976	\l__scripture_leftmargin_tl 307, 2313, 3937
\l__scripture_hanging_rightmargin_- tl 529, 2997	__scripture_list_para_end: 2052, 2059, 2100, 2108, 2374
__scripture_hanging_verse:n 1968, 2875, 2875, 2974	\l__scripture_main_language_tl . 301
__scripture_hanging_verse_- para:n 1949, 1949, 2973	\l__scripture_main_language_- variant_tl 304
\l__scripture_hanging_verse_- right_bool ... 533, 536, 2887, 2908	__scripture_mark_verse_format:nnn 310, 759
\l__scripture_hanging_verse_sep_- tl 539, 2881	__scripture_mid_para_chap_- begin:n 3129, 3129, 3905
__scripture_heading:n 2086, 2086, 2268	\l__scripture_mid_para_chap_bool 1762, 3050, 3146
\l__scripture_heading_aboveskip_- tl 226, 2101	\l__scripture_mid_para_chap_box 3080, 3147, 3189, 3193, 3200, 3213, 3215, 3221
\l__scripture_heading_afterindent_- bool 229, 2118, 2119	__scripture_mid_para_chap_end: 3156, 3156, 3908
\l__scripture_heading_align_tl 235, 236, 241, 246, 251, 2104	__scripture_mid_para_chap_end:n 3057, 3058, 3972, 4014
\l__scripture_heading_belowskip_- tl 258, 2126	__scripture_mid_para_chap_end_- orig:n 3057, 3057, 3060
\l__scripture_heading_font_tl 261, 2105	\l__scripture_mid_para_chap_- line_box 3051, 3089, 3092, 3160, 3163, 3184
__scripture_heading_format:n 264, 2106	\g__scripture_mid_para_chap_- prevdepth_dim 3052, 3159, 3225
\l__scripture_heading_show_bool 267, 269, 2090, 3947	\g__scripture_mid_para_chap_- prevgraf_int 2167, 2174, 2469, 2476, 3054, 3232
\l__scripture_indent_bool 272, 320, 1045, 1169, 2332	\l__scripture_mid_para_chap_- snap_box 3055, 3090, 3100, 3108, 3117, 3161, 3170, 3173
\l__scripture_inline_begin_tl 278, 2306	\l__scripture_mid_para_chap_- spacefactor_int .. 3056, 3086, 3109
\l__scripture_inline_bool 275, 2290, 2305, 2388, 3933	\l__scripture_mid_para_chap_- split_box 3080, 3199, 3203
\l__scripture_inline_end_tl 281, 2392	\l__scripture_mid_para_chap_- split_top_box 3080, 3202, 3205, 3208
\l__scripture_inline_font_tl 284, 2291	\l__scripture_mid_para_chap_- totalleftmargin_dim 3053, 3151, 3171
__scripture_inline_ref_format:n 288, 3939	__scripture_mid_para_drop_chap_- output: 1763, 3083, 3083
\l__scripture_inline_ref_sep_tl 291, 3940	\l__scripture_midparachap_show_- verse_bool 543, 546, 3122, 3142, 3696, 3704
\l__scripture_inline_version_- delim_tl 294, 3944	\g__scripture_mode_vertical_- inner_bool ... 775, 1127, 1211, 1229, 1238, 1982, 2230, 2458, 3991
__scripture_inline_version_- format:n 298, 3942	__scripture_name:n 768, 769, 770, 2140, 2140, 2279
__scripture_insert_verse_mark: 753, 753, 1545, 1631, 1916, 2824, 2927, 3126, 3553, 3591, 3630	
\l__scripture_language_tl 1388, 1392, 1396, 1401, 1405, 1410, 1417, 1424	


```

\l__scripture_name_font_tl 314, 2143
\__scripture_name_format:n 317, 2144
\l__scripture_narrow_aboveskip_-
  tl ..... 549, 3280, 3286
\__scripture_narrow_begin:n ....
  ..... 3237, 3237, 3911
\l__scripture_narrow_belowskip_-
  tl ..... 552, 3303, 3309
\__scripture_narrow_end: .....
  ..... 3298, 3298, 3914
\l__scripture_narrow_leftmargin_-
  tl ..... 555, 3261
\l__scripture_narrow_rightmargin_-
  tl ..... 558, 3262
\__scripture_no_first_verse: ...
  ..... 2147, 2147, 2282
\l__scripture_no_verse_first_-
  bool ..... 1853, 1873, 1876, 2149
\__scripture_nodrop_chap:n ....
  .. 1553, 1553, 2181, 2187, 2699, 2705
\l__scripture_nodrop_chap_coffin
  ..... 1437, 1568, 1609, 1618
\l__scripture_nodrop_chap_-
  voffset_dim .....
  .. 1438, 1576, 1580, 1585, 1613, 1622
\__scripture_nohang: .....
  ..... 2765, 2932, 2932, 2975
\__scripture_noindent_saved: 811,
  814, 1258, 1278, 1505, 1681, 1963,
  2054, 2066, 2102, 2115, 2192, 2289,
  2387, 2451, 2471, 2936, 3790, 3971
\__scripture_obeylines: .....
  ..... 118, 3666, 3668, 3929
\__scripture_obeylines_eol: ....
  ..... 3671, 3674, 3738
\l__scripture_outer_itemindent_-
  dim ..... 778, 2309,
  2310, 2317, 2534, 2536, 2548, 2619,
  2621, 2633, 2979, 2981, 3001, 3253,
  3255, 3269, 3274, 3786, 3788, 3810
\__scripture_para_chap:n .....
  ..... 1655, 1655,
  2180, 2186, 2698, 2704, 3358, 3364
\l__scripture_para_chap_indent_-
  dim ..... 1436, 1663, 1666,
  1667, 1670, 1672, 1673, 1674, 1684
\l__scripture_para_indent_tl ...
  ..... 451, 1338, 1933
\g__scripture_para_mode_vertical_-
  bool ..... 1287, 1291,
  1981, 2543, 2628, 2989, 3266, 3803
\__scripture_para_noindent: ...
  ..... 812, 1254,
  1254, 1499, 2063, 2112, 3961, 3982

\__scripture_para_par: .....
  ..... 61, 1289, 1289,
  1324, 2570, 2655, 3294, 3958, 3959
\l__scripture_parindent_tl ....
  ..... 322, 453, 685, 1230, 1292,
  1503, 2299, 2300, 2317, 2319, 2331
\l__scripture_parskip_tl .. 325, 2321
\l__scripture_pilcrow_bool ....
  ..... 328, 1983, 2094, 3751, 3795
\__scripture_pilcrow_output: ...
  ..... 800, 1128, 1133, 1212,
  1216, 1242, 1244, 1265, 1282, 1300,
  1302, 1506, 1633, 1770, 1976, 1979,
  1979, 2825, 2925, 3070, 3749, 3794
\l__scripture_pilcrow_sep_tl ...
  ..... 331, 1992
\l__scripture_poetry_aboveskip_-
  tl ..... 561, 3815, 3821
\__scripture_poetry_begin: ....
  ..... 3760, 3760, 3923
\l__scripture_poetry_belowskip_-
  tl ..... 564, 3838, 3844
\l__scripture_poetry_big_indent_-
  tl ..... 567, 3438, 3440,
  3490, 3495, 3536, 3539, 3661, 3663
\__scripture_poetry_ch:n .....
  ..... 3350, 3350, 3775
\__scripture_poetry_chap:n ....
  ..... 3355, 3368, 3368
\l__scripture_poetry_chap_-
  indent_dim .....
  .. 3322, 3385, 3389, 3399, 3405, 3428
\l__scripture_poetry_chap_-
  parshape_correction_dim 3324,
  3423, 3433, 3436, 3462, 3466,
  3480, 3485, 3511, 3515, 3527, 3532
\l__scripture_poetry_chap_sep_-
  dim 3322, 3393, 3402, 3417, 3426, 3442
\__scripture_poetry_end: .....
  ..... 3834, 3834, 3926
\g__scripture_poetry_end_bool ..
  ..... 2222, 3320, 3852, 3855
\l__scripture_poetry_indent_tl .
  ..... 570, 3390,
  3507, 3520, 3810, 3812, 3824, 3830
\l__scripture_poetry_leftmargin_-
  tl . 573, 3398, 3404, 3427, 3444,
  3459, 3470, 3506, 3519, 3627, 3806
\g__scripture_poetry_midparachap_-
  show_verse_bool ..... 3768
\l__scripture_poetry_midparachap_-
  show_verse_bool .....
  .. 3321, 3551, 3589, 3684, 3695, 3703

```

__scripture_poetry_mode_-		\l__scripture_ref_start_newline_-	
horizontal_verse:n	3637, 3637, 3686	bool	382, 385, 1012, 1022, 1028, 1034, 1066, 1083, 1100, 1111, 1122
__scripture_poetry_mode_-		\l__scripture_ref_start_sep_tl	.
vertical_verse:n			388, 950
.....	1726, 3383, 3451,	\l__scripture_ref_tl	
.....	3571, 3602, 3602, 3643, 3647, 3682,		942, 947, 949, 955, 966, 968,
.....	3715, 3725, 3732, 3740, 3744, 3777		1005, 1140, 2033, 2199, 2383, 2395,
__scripture_poetry_nodrop_-			2397, 2408, 2435, 2437, 2444, 2461
chap:n	 3359, 3365, 3556, 3556	\l__scripture_ref_vertical_sep_-	
__scripture_poetry_par:		tl	 391, 996
.....	3653, 3653,	__scripture_reference:n	
.....	3697, 3705, 3714, 3724, 3731, 3743		901, 901, 2441, 2444
\l__scripture_poetry_prevgraf_-		\l__scripture_reference_language_-	
int	 3325, 3449, 3656	tl	 360
\l__scripture_poetry_rightmargin_-		\l__scripture_reference_language_-	
tl	 576, 3807	variant_tl	 364
\l__scripture_poetry_secondhalf_-		__scripture_reference_start_-	
bool	 579, 3679, 3687	inline:	... 940, 940, 1024, 1030,
__scripture_poetry_text_-			1036, 1078, 1095, 1106, 1117, 1136
right:nn	... 3326, 3326, 3345, 3781	__scripture_reference_start_-	
\l__scripture_poetry_verse_-		newline:	
right_bool	 583, 585, 3613		953, 953, 1019, 1023, 1029,
\l__scripture_poetry_verse_sep_-			1035, 1067, 1084, 1101, 1112, 1124
tl	 588, 3605, 3640	__scripture_reference_start_-	
\g__scripture_post_env_indent_-		peek:	 999, 999, 3888, 3894,
box	 1288, 1314,		3900, 3906, 3912, 3924, 3973, 3996,
.....	1328, 1330, 1337, 1881, 1882, 2231		4002, 4008, 4015, 4021, 4027, 4050
\g__scripture_prev_inner_below_-		__scripture_reset_par_at_group_-	
skip	 777, 2232,	level:	
.....	2428, 2596, 2680, 3037, 3312, 3847	..	1773, 1778, 1778, 1816, 2514, 3235
\g__scripture_red_letter_active_-		__scripture_reset_spacing:	
bool	 1353, 1358, 1375		2013, 2021
\l__scripture_red_letter_bool	..	\l__scripture_rightmargin_tl	...
.....	334, 1356, 1372		394, 2314, 3938
\l__scripture_red_letter_colour_-		\l__scripture_selah_font_tl	...
tl	 337, 1359, 1365		400, 3344, 4037
__scripture_red_letter_off:	...	__scripture_selah_format:n	...
.....	1370, 1370, 2225, 2391, 2412		403, 3347, 4039
__scripture_red_letter_on:	...	__scripture_selah_output:	
.....	1354, 1354, 2224		3341, 3341, 3778
\l__scripture_ref_align_str	...	\l__scripture_selah_sep_tl	406, 3346
.....	344, 345, 346, 916, 923	\l__scripture_selah_text_tl	...
\l__scripture_ref_box			397, 3347, 4039
.....	880, 903, 909, 928, 938	__scripture_select_language:n	.
\l__scripture_ref_colour_tl	...		887, 895, 1386,
.....	350, 885, 886		1386, 1468, 1692, 1891, 2215, 3582
\l__scripture_ref_font_tl	. 354, 884	__scripture_set_pilcrow_hook:	.
__scripture_ref_format:n			796, 1200, 1205, 1246, 1248,
.....	357, 888, 3939		1280, 1304, 1306, 1924, 1930, 1952,
\l__scripture_ref_position_str	.		1971, 1971, 2273, 2945, 3072, 3074
.....	372, 373, 374, 1004, 2364, 2381, 2459	__scripture_setup_list_noindent:n	
\l__scripture_ref_sep_tl			779, 779, 2331,
.....	378, 910, 929, 935, 2401, 2406, 3940		2562, 2647, 3016, 3270, 3275, 3824

__scripture_setup_reset_-		\l__scripture_verse_par_start_-	
defaults:	218, 727	bool	1852, 1859, 1860, 1864, 1875, 1895
__scripture_setup_style:nnm ...		\l__scripture_verse_para_bool ..	448, 786, 808,
.....	731, 731, 737, 746, 3868		833, 981, 1276, 1334, 1475, 1546,
__scripture_skip_diff:NN	818, 818,		1558, 1765, 1879, 1986, 2061, 2110,
2327, 2422, 2558, 2592, 2643, 2676,			2464, 2568, 2653, 2826, 2972, 3293,
3012, 3033, 3285, 3308, 3820, 3843			3747, 3750, 3796, 3955, 3983, 4054
\g__scripture_suppress_next_-		__scripture_verse_para_output:n	1927, 1927, 3957
pilcrow_bool	957, 958, 960, 1323, 1333,	\l__scripture_verse_presep_tl ..	454, 1865, 1869, 2293
.....	1491, 1650, 1678, 1679, 1683, 1729,	\l__scripture_verse_sep_tl	457, 1911, 2879, 2881, 2928, 3605, 3640
1730, 1734, 1738, 1959, 1970, 1988,		\l__scripture_verse_show_bool ..	439, 460, 1885
1989, 2227, 2228, 2544, 2599, 2629,		\l__scripture_version_delim_tl .	466, 894, 3944
2683, 2691, 2884, 2906, 2990, 3040,		__scripture_version_format:n ..	469, 896, 3942
3273, 3315, 3633, 3804, 3977			472
\g__scripture_suppress_next_-		\l__scripture_version_language_-	476
verse_para_bool	795, 1264, 1279, 1854,	tl	476
.....	1935, 1936, 1953, 1967, 2229, 2944	\l__scripture_version_language_-	482, 483, 484, 944, 963, 1141, 2384
__scripture_temp_ch	1069	variant_tl	463, 905, 947,
__scripture_temp_ch:	1076		966, 1142, 2385, 2399, 2402, 2408, 2439
__scripture_temp_newch: .	1086, 1093	\l__scripture_X_char_tl	65,
__scripture_temp_newvs: .	1114, 1116		215, 1074, 1091, 1441, 2239, 2968, 3773
__scripture_temp_vs:	1103, 1105	__scripture_zero_chap_par_-	
__scripture_text_right:nn		prevgraf_at_group_level:	1839, 1839, 1849, 3372
.....	851, 851, 2284		12, 49
\l__scripture_text_right_sep_tl	414, 2284, 3782	scripture/chap/after (hook)	12, 49
__scripture_vbox_set_top:Nw ...		scripture/chap/before (hook)	13, 49
.....	3044, 3044, 3147	scripture/heading (mark)	49
\l__scripture_verse_colour_tl ..		scripture/pilcrow (hook)	105
.....	420, 1889, 1890	scripture/poetry/para/after (hook) .	49
__scripture_verse_compact_-		scripture/poetry/pilcrow (hook)	13, 49
output:n	1921, 1921, 3989	scripture/verse (mark)	12, 49
\l__scripture_verse_first_bool .		scripture/verse/after (hook)	12, 49
.....	424, 1894, 2295	scripture/verse/before (hook) ...	12, 49
__scripture_verse_first_-		\scripturecurrentbook	755, 760, 2209, 3859
format:n	427, 1904, 2297	\scripturecurrentchapter	8,
\l__scripture_verse_first_sep_tl			761, 1447, 1555, 1657, 2210, 3558, 3859
.....	430, 1906	\scripturecurrentverse .	8, 762, 1448,
\l__scripture_verse_font_tl	433, 1888		1556, 1658, 1857, 2212, 3559, 3859
__scripture_verse_format:n		\scripturesetup	8, 3862
.....	436, 1909, 2297	\scripturestyle	8, 27, 3866
\g__scripture_verse_int		\scripturestyle*	8
... ..	417, 2203, 2213, 2249, 2263, 2265	\scshape	22, 316, 650
\l__scripture_verse_language_tl	441	\section	3, 6
\l__scripture_verse_language_-			
variant_tl	445		
__scripture_verse_output:n			
.....	1543, 1629, 1855, 1855, 1925,		
1947, 2243, 2822, 2891, 2899, 2912,			
2920, 2930, 3111, 3124, 3552, 3590,			
3610, 3617, 3625, 3635, 3651, 3746			

<code>\selah</code>	8, 3778, 4034	1175, 1180, 1239, 1701, 1702, 1703, 1899, 2342, 2346, 2350, 2359, 2363
<code>selah/font</code> (option)	25	
<code>selah/format</code> (option)	26	<code>\str_if_exist:NTF</code>
<code>selah/sep</code> (option)	26	344, 372, 482
<code>selah/text</code> (option)	26	<code>\str_new:N</code>
<code>\selectfont</code>	1465	345, 373, 483, 2032
<code>\selectlanguage</code> ..	21, 22, 1396, 1405, 1408	<code>\str_set:Nn</code>
<code>\setbox</code>	3046	2200, 2525, 2610, 2957, 3131, 3244, 3767, 3970, 3995, 4001, 4007, 4013, 4020, 4026
skip commands:		<code>\str_set_eq:NN</code>
<code>\skip_eval:n</code>	820, 1496, 1592, 2754, 3379, 3567	346, 374, 484
<code>\skip_gset_eq:NN</code>	2596, 2680, 3037, 3312, 3847	<code>\strut</code>
<code>\skip_gzero:N</code>	2232	859, 974, 2886, 2907, 3087, 3607
<code>\skip_horizontal:N</code>	950, 1625, 1684, 1756, 1914, 1994, 2401, 2406, 2809, 2901, 2922, 3229, 3417, 3585, 3627, 3648	<code>style</code> (option)
<code>\skip_horizontal:n</code>	785, 869, 934, 962, 1503, 1599, 1803, 1869, 2049, 2097, 2782, 2785, 2935, 3116, 3179, 3332, 3646	26
<code>\skip_if_eq:nnTF</code>	1492, 1588, 1865, 2293, 2750, 3375, 3563	
<code>\skip_new:N</code>	777	
<code>\skip_set:Nn</code>	2002, 2321, 2325, 2420, 2556, 2590, 2641, 2674, 3010, 3031, 3283, 3306, 3818, 3841	
<code>\skip_set_eq:NN</code>	2003, 2550, 2635, 3004, 3277	
<code>\skip_vertical:N</code>	915, 920	
<code>\skip_vertical:n</code> ...	2029, 2428, 3227	
<code>\skip_zero:N</code>	874, 1660, 2004, 2005, 2320, 2323, 2413, 2418, 2551, 2554, 2588, 2636, 2639, 2672, 3005, 3008, 3029, 3118, 3154, 3278, 3281, 3304, 3337, 3813, 3814, 3816, 3839	
<code>\c_zero_skip</code>	33, 48, 109, 596, 601, 611, 1492, 1588, 1625, 1865, 1914, 1994, 2293, 2750, 3375, 3563	
<code>\small</code>	19, 263, 634	
<code>\smash</code>	17, 1606	
<code>\spacefactor</code>	3086, 3109	
str commands:		
<code>\str_case:nn</code>	65, 115, 194, 237	
<code>\str_if_eq:nnTF</code>	799, 810, 885, 916, 923, 944, 961, 963, 1131, 1141, 1215, 1346, 1360, 1376, 1466, 1477, 1570, 1632, 1664, 1680, 1689, 1737, 1769, 1889, 2043, 2088, 2303, 2381, 2384, 2459, 3580	
<code>\str_if_eq_p:nn</code>	170, 854, 855, 970, 971, 984, 988, 1004, 1008, 1009, 1051, 1056, 1175, 1180, 1239, 1701, 1702, 1703, 1899, 2342, 2346, 2350, 2359, 2363	
		<code>\T</code>
		$\mathrm{T}_{\mathrm{E}}\mathrm{X}$ and $\mathrm{L}^{\mathrm{A}}\mathrm{T}_{\mathrm{E}}\mathrm{X} 2_{\epsilon}$ commands:
		<code>\@afterheading</code> ..
		995, 1712, 2076, 2125
		<code>\@beginparpenalty</code>
		2220
		<code>\@clubpenalty</code>
		1319
		<code>\@doendpe</code>
		62, 3960
		<code>\@endparpenalty</code>
		2221
		<code>\@ifpackageloaded</code>
		1394, 1399
		<code>\@lowpenalty</code>
		2219, 2220, 2221
		<code>\@nolnerr</code>
		2575, 2660, 3966
		<code>\@outerparskip</code>
		2417, 2423, 2587, 2593, 2671, 2677, 3028, 3034, 3303, 3309, 3838, 3844
		<code>\@par</code>
		2374, 3959
		<code>\@restorepar</code>
		1318
		<code>\@topsepadd</code>
		2418, 2420, 2588, 2590, 2596, 2672, 2674, 2680, 3029, 3031, 3037, 3304, 3306, 3312, 3839, 3841, 3847
		<code>\@totalleftmargin</code>
		99, 1512, 1524, 1534, 1641, 1814, 1943, 1961, 2025, 2045, 2092, 2416, 2446, 2483, 2486, 2496, 2585, 2601, 2669, 2685, 2727, 2737, 2745, 2856, 2865, 2867, 2869, 2871, 3042, 3151, 3152, 3317, 3433, 3438, 3479, 3489, 3526, 3536, 3658, 3661, 3856
		<code>\f@size</code>
		1454
		<code>\textright</code>
		8, 2283, 3779, 4043
		<code>textright/sep</code> (option)
		26
		<code>\textscripture</code>
		9, 3870
		<code>\textsuperscript</code>
		27, 438, 681
		tl commands:
		<code>\c_space_tl</code>
		19, 21, 296, 468, 644, 690, 3110, 4038, 4040, 4046
		<code>\tl_clear:N</code>
		1140, 1142, 2383, 2385
		<code>\tl_gclear:N</code>
		2204, 2461
		<code>\tl_gset:Nn</code>
		741, 1447, 1448, 1555, 1556, 1657, 1658, 1857, 2209, 2210, 2212, 3558, 3559
		<code>\tl_if_empty:NTF</code>
		755, 942, 955, 1392, 1403, 2399, 2439
		<code>\tl_if_empty:nTF</code>
		891, 893

