Bible Modules

- How to use Bible Modules in Paratext
- Troubleshooting Modules
- Example Modules
- Advanced Bible Modules
- Bible-Module-Repository
  - Submit a Bible Module

What is a Bible Module?

A Bible Module is a document containing Scripture references, where Paratext inserts the text of those references from the current project. While you can copy and paste text from your Paratext project into another document, a Bible module makes this easier. Instead of finding each reference in the Scriptures to copy, you can just enter a reference and Paratext looks it up for you. Also, Paratext will refresh the content in your document if the cited texts are updated.

Examples of a Bible module:

- A script for a media production (which might break the text of a book into the verse ranges which will be recorded as audio clips).
- A lectionary (which has verse ranges from various books for the selected readings in the worship services).
- A Bible story book with a mixture of Scriptures and added narrative.

A Bible module consists of a template with text and Biblical references in any order for creating a Scripture-based product. You view a Bible Module in an Extra book (XXA - XXF). When activated, the Bible Module will replace all of the Scripture references with the actual text from the active project. The resulting file can be printed or exported as any other Scripture book in Paratext.

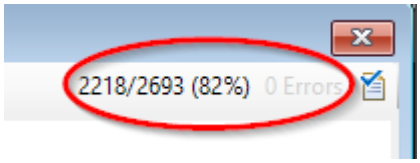
Bible Modules allow for:

- Automated additions or replacements to the Scripture content
  - E.g. Replace a pronoun with a name when the original context is missing.
- Discontiguous ranges of verses

You view a Bible module in an Extra book. From the **View** menu you may view the document as

- Unformatted Specification
  Raw text, similar to unformatted mode in a normal project window.
- Standard Specification
  Formatted specification text, without pulling and displaying the referenced text
- Standard Output
  Formatted specification text and backslash codes with actual reference text displayed in place of the references.
- Preview Output
  Formatted specification text with actual reference text displayed in place of the references.

At the top of the view, there is a status bar that shows how many of the referenced verses are found in the project.



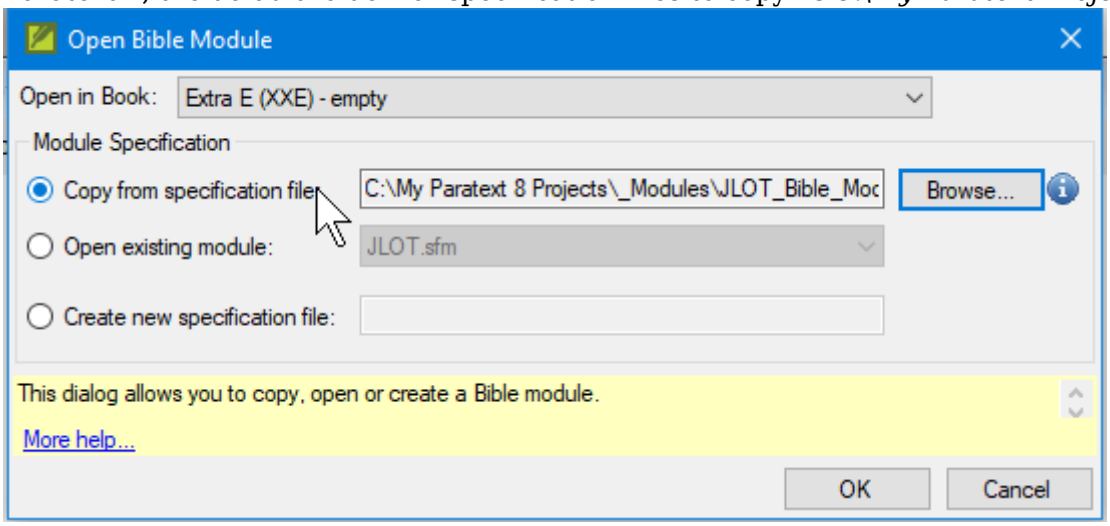
Clicking on the symbol of the document with the blue check mark will present you with a list of verses that are missing in the project, and a list of any reference errors in the module. PLEASE NOTE: In the current versions of Paratext 7.6 and 8.0, Empty books with chapter and

verse markers included will be counted as translated text. Paratext only checks to see if the verse exists, but not if there is text in the verse. Hopefully this will be fixed at some point. It is best to create empty books without chapters and verses, adding the verse numbers as you translate with Ctrl-K.

Creating a Bible module

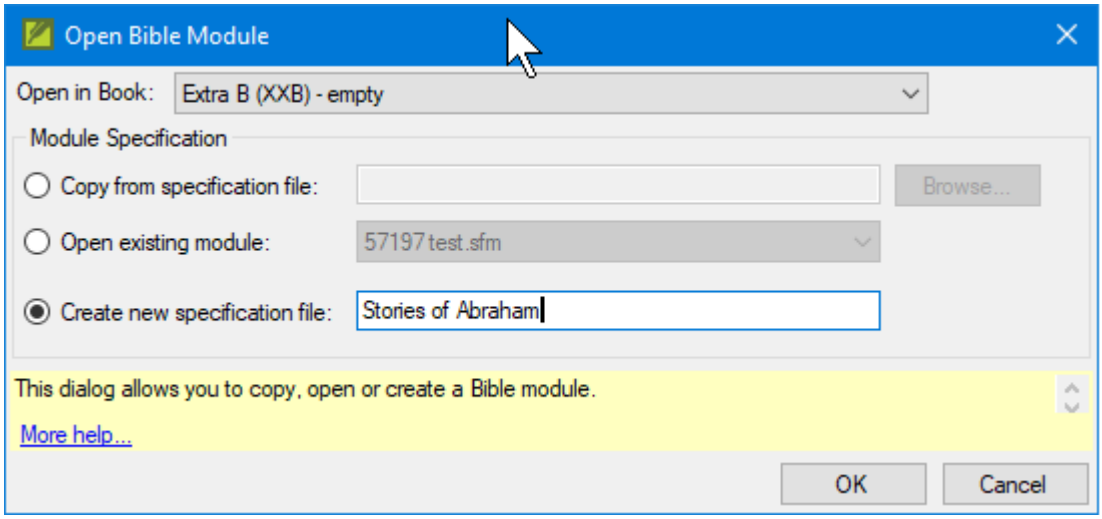
Paratext does not have any built-in specification files or templates. There are some example Bible modules you can download on another page. Or you can attempt to create a module from scratch.

In Paratext, go to **Tools > Open Bible Module.** In the dialog, select which extra book (XXA to XXE) you want Paratext to use for it. To use a template or specification file someone else created, choose the **Copy from specification file** option, then select the file. The default folder for these is *C:\My Paratext 8 Projects_Modules*, but you can use the browse button to select the file anywhere on your computer. (In Paratext 7, the default folder for specification files to copy is *C:\My Paratext Projects\Modules).*



Paratext will create a new module file in the project folder. You can edit this file, and your edits will not affect the original file.

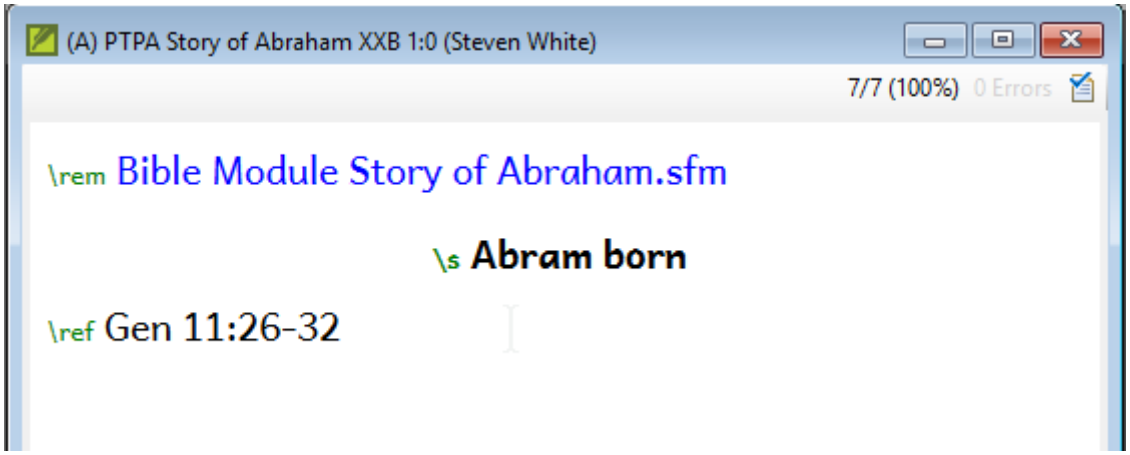
You can also create a new specification file by choosing that option, and entering a new name for the file. For example here we are starting a new module file, Stories of Abraham.



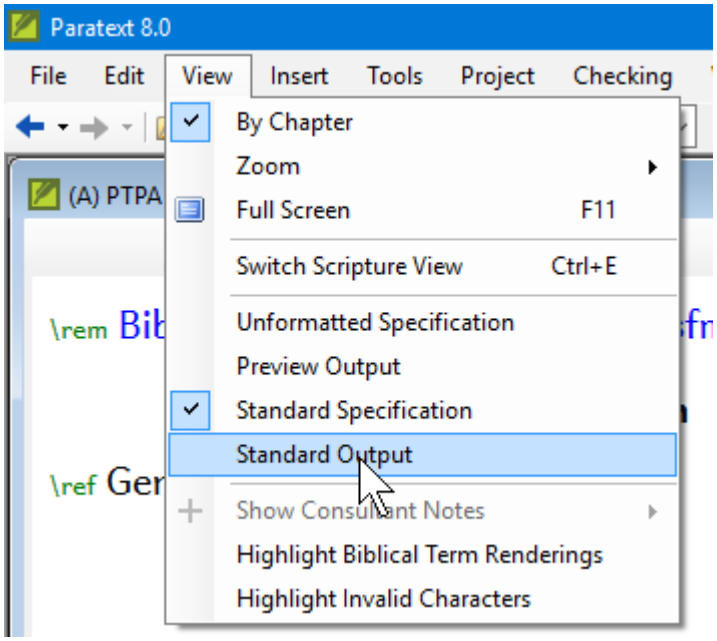
You can then edit it in Paratext. Paratext saves this specification file inside the *Modules* subfolder of the project folder.

For example, we could start filling in our Stories of Abraham module by adding text such as titles, and adding reference codes for Paratext to fill in with the appropriate text.

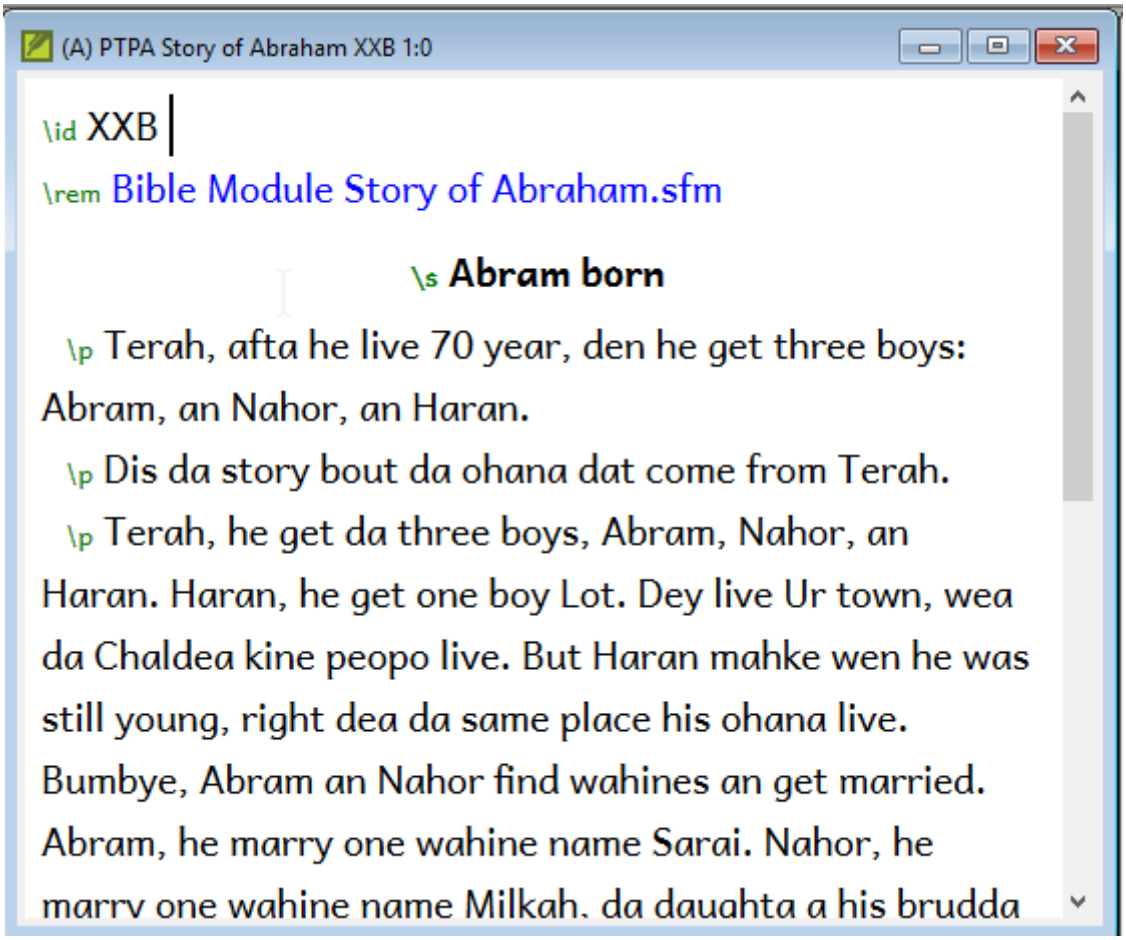
If we type this into the module file:



then in the view menu switch to **Standard output**



Paratext will insert the specified text from the project.



You can also create a specification file outside Paratext, if you save it as UTF8 text and follow the restrictions for naming the file: The name can contain only the following: letters (A-Z a-z), digits (0-9), underscore, or space.

Special markers in module files

The specification includes lines which begin with the following special markers.

Marker	Description
\vrs	Identify the versification of the references in the specification. Options: English, Septuagint, Original, Russian Protestant, Russian Orthodox, Vulgate Example: \vrs English
\inc	Include certain markers (and their text) which are excluded from the content by default. Options: v – verse numbers f – footnotes x – cross references s – section headings Example: \inc v
\ref	Copy the markers and text for a verse or verse range from a project. This reference format is limited: it allows hyphen and comma (without a following space) but does not allow semicolon. Example: \ref PSA 72:1-7,18-19
\refnp	Copy the verse range with no paragraph markers from the project. Example: \p \refnp LUK 1:1-4
\rep	Find text within a verse range in an ordinary book of a project and replace it with other text in the content of the module. To separate the text to find from the text to replace, type an equal sign and a greater than sign. If you need to prevent a replacement from occurring more than once, include enough surrounding context so that the text to find is unique. Example: \rep he=>Jesus

\mod	Include the content of a child module in the content of this (parent) module. Example: \mod child_module.sfm

In addition to extracting a section of text to display in the module, you can also display the reference using the book names and format of the project's Scripture reference settings.

Displaying reference names in the target language

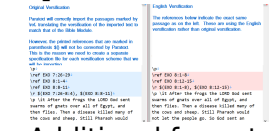
You can use codes to have Paratext give the reference using the book names as defined in the project settings. The code pattern is **$(BBB C:V)** where **BBB** is the three-character book abbreviation, **C** is the chapter number and **V** is the verse number. (A formatted reference can occur anywhere in the specification after the \id line. For example, in \sr section range references.) This reference format does not have the limitations of the \ref format. The reference:

- must be surrounded by parentheses
- must use the standard three-character book abbreviation
- must use the default punctuation for references (a colon to separate the chapter number and verse number, and a hyphen to separate verse numbers).

When the reference is copied to the Bible module; the standard three-character book abbreviation is replaced by the choice for cross references on the Book Names tab of the Scripture Reference Settings dialog. The default punctuation is replaced by the punctuation specified on the Reference Format tab of the Scripture Reference Settings dialog. For example, in a project where the cross reference choice is to use the book abbreviation; where the book abbreviation for the book of Revelation is Ap; where the chapter verses separator is a period, and where the range of verses separator is a hyphen, $(REV 1:1-16) will become Ap 1.1-16 in the content of the module.To override this default behavior, use any of the following:

- $a(BBB C:V) to use the book abbreviation
- $s(BBB C:V) to use the book short name
- $l(BBB C:V) to use the book long name

Versification differences are not translated by the display references notation $(). If you need to display the reference text for texts of different versification schemes, you will have to create a separate specification file for each versification. Click on the image below to see an example.



Additional format codes

You may also use format markers from the project stylesheet. For example, \s (headings), \p (paragraphs), and \rem (comments)

These codes and their following text will be formatted and copied to the output.

Note

- The \refnp marker removes all paragraph markers from the copied text. Therefore, if you want to use \refnp to copy a range of verses, you might want to break the range up into several smaller ranges of verses and precede each \refnp with a \p marker to break the verses into paragraphs manually.
- The \refnp marker can produce some unusual results when used with the \inc s option.
- Any chapter numbers are excluded from the copied project text. If the content needs chapter numbers, for each chapter number needed, the specification itself must include the \c marker followed by the chapter number. However, if the specification contains duplicate chapter numbers (for example, corresponding to multiple books), the View menu options will not display the content.

Video Tutorial

- **A how to video: https://vimeo.com/185916082**
  - Note: This method uses an older export to RTF method. We now recommend using **PTXprint to print/publish your Bible modules**.

Further Information

- Troubleshooting Modules
- Example Bible Modules