Key Bloom Concepts

What shellbook makers, advisors, and trainers should know

[Last updated for version 3.0]

Bloom is designed to help low-literacy communities develop a library of books in their own language. Bloom keeps book production simple and efficient. You can either make your own books using your own texts and graphics or you can translate a ready-made book from a source language.

By now, you've probably played around with the program, and hopefully you found most of it self-explanatory. But you have some questions. And hopefully you're curious to understand how all the pieces fit together, so that you can confidently make decisions and train others. In this document, we'll look at the key components of Bloom. Some good information is covered elsewhere:

- The goals of Bloom, who is trying to reach, etc. are covered on the Bloom web site
- Instructions on how to do things can be found via the "Help: Documentation" menu
- The file format documentation is available for those *very few* technicians who need to know it.

In this paper we're aiming for a deep understanding of how Bloom differs from a normal Page Layout program.

Bloom is about books, so let's start by figuring out what Bloom's idea of *book* is.

Book

In Bloom, the term *book* is a generic one meaning "Bloom document". Bloom *Collections* are made up of *books*, which could be any of the following:

- Original collection of folk stories from that culture
- A guide to a common disease, translated into the vernacular from a *shell book*
- A wall calendar
- A picture dictionary

In Bloom itself, books are represented by little icons with a title for the caption.

See also: Front Matter, Illustrations, Page Size and Orientation, Source Texts, Books On Disk

Book Sources: Templates and Shell Books

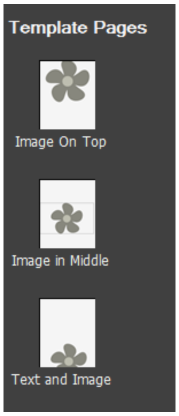
In most word processors and desktop publishing applications, you start documents with a *blank slate.* They may provide template documents too, but either way you have full control over every conceivable aspect of the document, from where the text sits on the page to whether a watermark shows in the background.

That's not what Bloom is about. In Bloom, you trade some of that flexibility for simplicity and productivity. Instead, a new Bloom book is always based upon one of two kinds of other books: a *template*, or a *shell book*.

Templates

A Bloom *Template* contains one or more *template pages*. A template page has no content; it just gives you places to type and or places to place illustrations.

In Bloom 3, there are six book templates you can use to start a new book:


Template Pages

- Basic Book: has the simple page layouts (illustrated at right) that are used in the majority of existing Shell Books. Supports multiple Page Sizes and Orientations. Includes a "Custom" page that lets you divide the page up into blocks of text and pictures.
- Decodable Reader: Helps you create books as part of a series in which you carefully control what letters are introduced at each stage.
- Leveled Reader: Helps you stick to your desired reading level by monitoring the number of words in each sentence an on the page.
- Picture Dictionary: currently has a single template page that has six "Word and Picture" blocks. You can drag these around and resize them. Supports monolingual, bilingual, and trilingual dictionaries. [experimental]
- Wall Calendar: you enter in what year you want and give the vernacular terms for months and days. It then creates an A4 Landscape calendar with places to enter a large illustration, short text for the month, and daily events.
- Story Primer: This template, which you can get on BloomLibrary here, has many template pages for making primers.

If you can't make the book you need using the Basic Book and Custom Page template, do write and let us know.

Shell Books

You've probably seen a Shell Book prepared in a program like PageMaker, Publisher, or Open Office. These typically have:

1. Source texts which come as separate files
2. A list of pages laid out for a certain page and orientation
3. Illustrations
4. Places to type in the vernacular
5. Some "Front matter" containing acknowledgments, copyright and perhaps a license for re-use

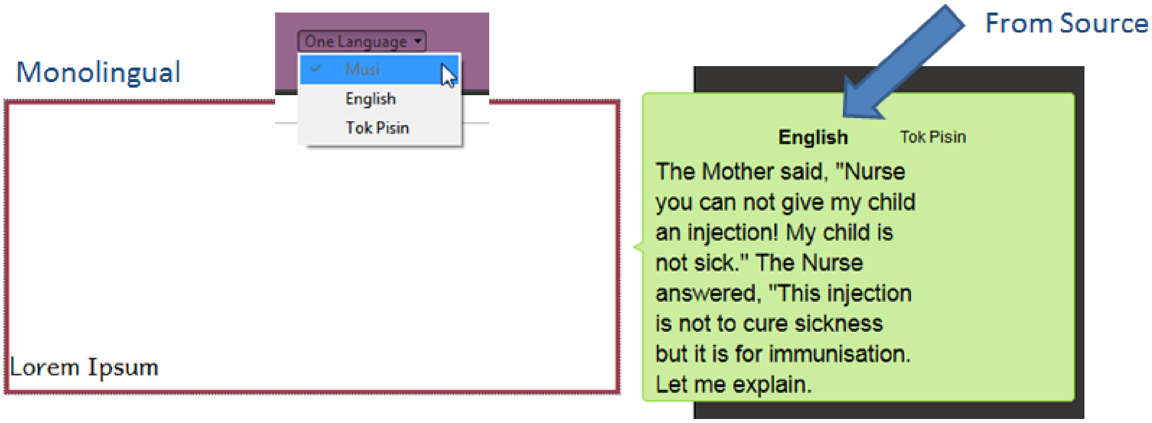
Bloom Shell Books have all these same components, but every one of them is done in a way that furthers its two core goals, which are to

A. “Lower the Bar” on how skilled you need to be to make books in your own language
B. Promote a growing worldwide ecology of unencumbered Shell Books.

Here are some highlights:

Source Texts

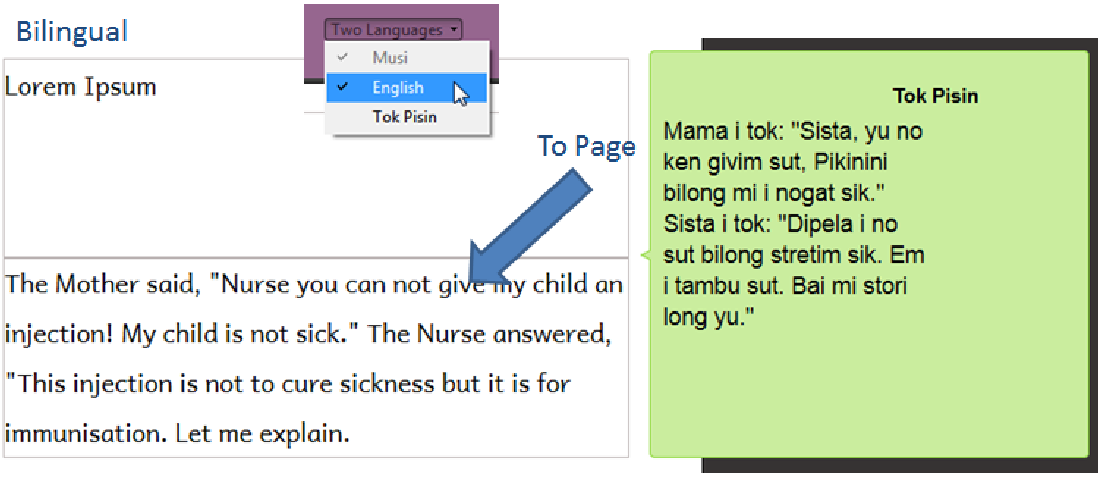
Generally, the text you translate from, what we call a *source text*, comes in a separate text file. So you might have a text file that gives you the English source, and another file that provides the Tok Pisin source or some other major language. In Bloom, ***all source texts come built-into the book.*** In fact, when someone takes your shell book and translates it into the vernacular, Bloom simply adds that vernacular to the list of texts in the book. Why is this good? Imagine that a user first just translates a Shell Book into the vernacular, using English as a source language:



Lorem Ipsum

The Mother said, "Nurse you can not give my child an injection! My child is not sick." The Nurse answered, "This injection is not to cure sickness but it is for immunisation. Let me explain.

Now consider what happens later when they choose to make a diglot version of the book, showing the English. Because the book’s file actually contains all those other languages, all Bloom has to do is dig into the list of texts, pull out the English, and display it alongside the vernacular instead of in the source texts box:



Lorem Ipsum

The Mother said, "Nurse you can not give my child an injection! My child is not sick." The Nurse answered, "This injection is not to cure sickness but it is for immunisation. Let me explain.

Mama i tok: "Sista, yu no ken givim sut, Pikinini bilong mi i nogat sik." Sista i tok: "Dipela i no sut bilong stretim sik. Em i tambu sut. Bai mi stori long yu."

> *Essentially, each Bloom book is a little database of texts in various translations, ready to be displayed either in the prompting box, or in the text itself.*
>
> *An interesting implication of this design is that any Vernacular book can become a shell book for other languages; the formats of a shell book and vernacular book are identical.*

Page Size & Orientation

In Bloom, templates can be designed to **automatically resize and rearrange** to fit different page sizes and orientations.

Licensing

Ten years ago, few of us talked about “licensing”. We thought “copyright” was what was important. Now, we realize the opposite is true. **Who holds the copyright is often irrelevant**. What counts is what permissions they give you (the *license*). So Bloom offers built-in help for understanding, picking, and displaying a license. It even lets you do that for illustrations, and goes so far as to embed that copyright, illustrator name, and license in the image file itself, so it can’t be lost.

Vernacular Collection

Bloom calls a group of books in a local language a *Vernacular Collection*. When you first run Bloom, it asks you if you want to make one, and ever after, it will open up showing the contents of that collection in the upper left corner. In the lower left corner, Bloom lists all the *templates* and *shell books* you can use to add books to your Vernacular Collection.

Languages of Vernacular Collections

Naturally, each Vernacular Collection has a primary language, simply called “the vernacular”.

1. Vernacular

In addition, each collection knows about up to two languages of wider communication:

2. Language 2
   - National or other language of wider communication
3. Language 3
   - Regional or other language of wider communication
   - Optional

So what does Bloom do with these settings? Vernacular is used for the primary title on the cover and title page, and for the primary contents on the inner pages. Language 2 & 3 also show up in the front matter (cover, title, verso page). Beyond that, they are used if you set the book to “two languages” (bilingual) or “three languages” (trilingual).

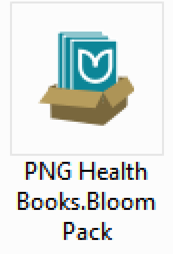
See also: Vernacular Collections On Disk

Source Collections

Source Collections look and act very much like Vernacular Collections. The difference is their purpose. Instead of being a collection of works in a local language, they are a collection of Shell Books & Templates waiting to be translated into various local languages around the world.

Sharing Source Book Collections

We have a pretty simple way of sharing Source Collections with others. We package them up in a single file, called *BloomPack*. This is actually just a .zip file, renamed. When you double click on it, it launches Bloom, which copies the collection inside to that machine's Bloom Collections directory[1].



Sharing Source Collections

Ready to share beyond your nearby colleagues[2]? For now, here's our low-budget way to share:

1) Create a Dropbox account
2) Save your BloomPack to the Dropbox folder on your computer
3) Right click that file, and choose "Get a Link…"
4) Email that link to me (hattonjohn at gmail com), and I'll share it on the Bloom site, along with a description of your collection.

Adding New Languages To Existing Source Collections

A promising thing about Bloom Shell Books is that we can put dozens of source texts in a single Shell Book. The user translating into vernacular just looks at the one(s) that she reads best. How to do that?

1) Find out who maintains that collection, tell them what you want to do, so that no one else does it at the same time.
2) Locate the Shell Collection you want to work with on the disk, and open it.
3) Under "Settings", set Language 1 to the language you want to add.
4) Restart Bloom.
5) Now open one of the shell books, and translate all the text into your Language, just like you would for a Vernacular Book.
6) When you're done, click on the Make Bloom Pack button, and name it something like "Health Books with French.bloomPack" and send it back to whoever maintains the collection, so that they can re-publish it.

In the future, we can imagine both Send/Receive ways of doing this collaboration, and more formal server-based ways. In any case, for now, we can get by with sharing and collaborating the old fashioned way ☺

[1] On Windows 7, that's c:\programdata\sil\bloom\collections

[2] There are long-term plans for DSPACE (e.g. SIL REAP)-based repository. But for now, this should work.

Front/Back Matter

By *Front Matter*, we mean the cover, title page, and credits page[3]. Similarly, *Back Matter* are those pages which come after the content, including the back cover. When creating a Shell Book, you'll fill in most of the content of these pages, things like:

- Title in one or more languages
- Cover illustration
- Book topic
- Credits to the author(s) and illustrator(s), copyright, and license
- ISBN # of the shell itself

When someone makes a vernacular book from your Shell Book, they will add

- A vernacular title
- Acknowledgements for the translator(s)
- (optionally) ISBN # of the vernacular book

Let's look at what Bloom does special to help everyone do the right things in the front matter. Bloom starts by making it clear what fields you should fill in/change depending on what you're doing, be it creating a Shell Book, or translating a Shell Book into the vernacular. So when you're creating a Shell Book, Bloom doesn't let you fill in the box for acknowledgments for the translators, and when you're translating a Shell Book into the vernacular, you can't change copyright, original acknowledgments, etc.

Title

Now let's go through each of the components in the front matter, starting with the **Title**. When you create a Shell Book, let's say you enter the title, on the cover, in English. If you now switch to the Title Page, you'll see that Bloom has already copied the same title over there, where it goes. Now if you were to make a vernacular book using this shell, you'd notice that Bloom automatically took your English title, big and bold, and moved it down to in smaller type below the vernacular, which now takes the primary spot on the cover.

Topic

For the **Book Topic**, Bloom intentionally constrains you to a list of approved topics[4]. Bloom does this because if everyone came up with their own topics, it would get hard to find things, and like books wouldn't end up in the same topic.

Copyright & License

It's *vital* that you understand the difference between copyright and license. Here's how I think about the most relevant aspects of the issue:

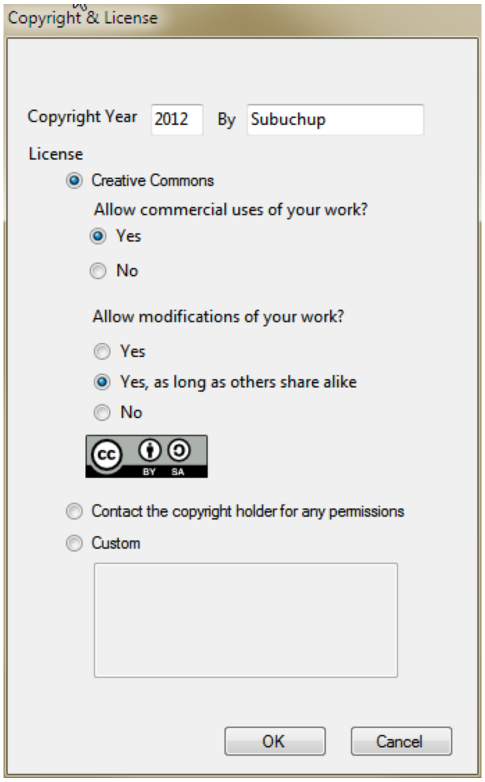
License: What you're allowed to do with the material

[3] In the future, this will also include the government approval letters required in some countries.

[4] Ideally, some group like the SIL International Literacy Department will take responsibility of maintaining this list.

Copyright: Who claims the right to issue the license(s)

For the copyright & license, Bloom actually opens up a small dialog box that helps you think about what you need to say. Ideally, you'll give it a nice Creative Commons license, but for some legacy materials, you can paste in a custom license.



Copyright & License

Copyright Year 2012 By Subuchup

License

● Creative Commons

Allow commercial uses of your work?

● Yes

○ No

Allow modifications of your work?

○ Yes

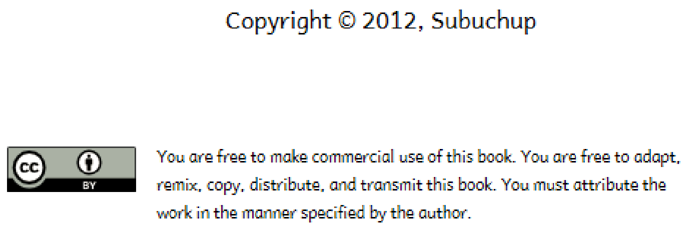
● Yes, as long as others share alike

○ No

○ Contact the copyright holder for any permissions

○ Custom

It then creates a section displaying what you said. If you chose a Creative Commons license, it displays a paragraph explaining the license in simple language:

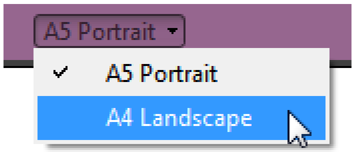


Copyright © 2012, Subuchup

You are free to make commercial use of this book. You are free to adapt, remix, copy, distribute, and transmit this book. You must attribute the work in the manner specified by the author.

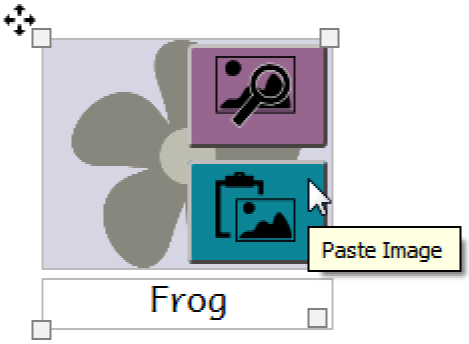
Page Size and Orientation

Imagine that you first create a book for printing as small booklets, but then later want to print some copies large, and landscape, so that a teacher can



display them in front of a classroom. This isn't just a matter of changing the printer settings; illustrations and text often need to be shifted around. In Bloom, templates can be designed to **do that arranging for you, automatically**.[5]

Illustrations



Bloom has a couple special features related to **illustrations.** First, each place that can have an illustration initially comes with a placeholder image. When you hover over it, two buttons appear. The "Paste" button just pastes the image on your clipboard in place of the placeholder. The "Change" button brings up the Palaso Image Toolbox, which lets you pick pictures from Art Of Reading, your camera, scanner, or your hard drive. Whichever you choose, Bloom automatically scales the illustration to fit the template.[6]

Bloom emerged at a time of rapidly increasing awareness and sensitivity to the rights of the contributors. In years past, Sally (a foreign NGO worker) would get Joe (a community member) to make illustrations, they'd put out a shell book, and everyone was happy. Joe understood that there was no money involved here. Fast forward to the present day when then NGO wants to put the book on the web for the benefit of the community, and alarms go off. Did Joe approve of this? Did he agree to let other people use his drawings?

A related problem is the use of illustrations that clearly are not from the community, but are instead of unknown origins.

"Bill, how do you know you can use that illustration in your Picture Dictionary?"

"Because Sharon said it is OK".

See the problem? Bloom takes a few steps towards helping the situation. In the Copyright & License section above, we saw how Bloom helps gather this information. For books, it just prints the information on the Title Page. But for illustrations, it embeds this information, along with the illustrator name, *in the image file itself* using the Adobe Extensible Metadata Platform (XMP). So even if you look on the disk, find that .png file, and take it somewhere else, this information is still retrievable.

[5] How does it do that, and to what extent can it re-arrange? Bloom style sheets tell Bloom what Size & Orientations they support. Bloom, in turn, tells the style sheet what the user chose. The style sheet can then set the size and location of each item. The mono/bi/trilingual feature works the same way: the style sheet is told which mode we're in, and it can then rearrange and resize items accordingly.

[6] See the Illustrations section for a discussion of how Bloom also helps track where that picture came from and what the license of it is.

Bloom embeds Illustrator, Copyright, and License information in the image file itself.

The View from the Hard Drive

The normal user of Bloom might never look on disk and see where the Vernacular Collection lives, because Bloom goes to great lengths to make that unnecessary. However, it's nice to know where the files and how they are organized, in case you need to get at them.

Vernacular Collections on Disk

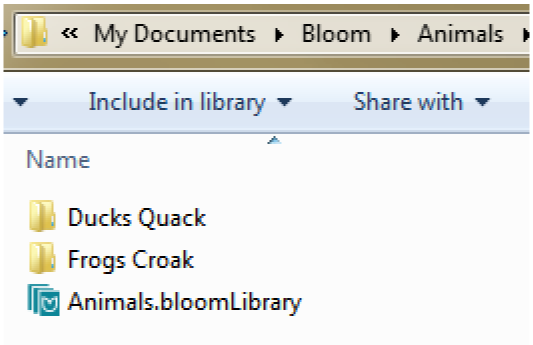
Vernacular collections consist of a folder, created under My Documents/Bloom. That folder contains:

- a single settings file (.BloomCollection)
- a *Cascading Style Sheet* named "collection.css" which you can use to override various text settings or the entire collection (e.g. change the default font)
- and one folder per book.

You're welcome to copy this folder to other computers, or just move it to a location you prefer. If you do, you can show Bloom where you put it.

Source Collections on Disk

On disk, Source Collections look exactly like Vernacular Collections. The only difference is to be found if you open up the .BloomCollection file; you'll see a setting identifying this as a Source Collection.

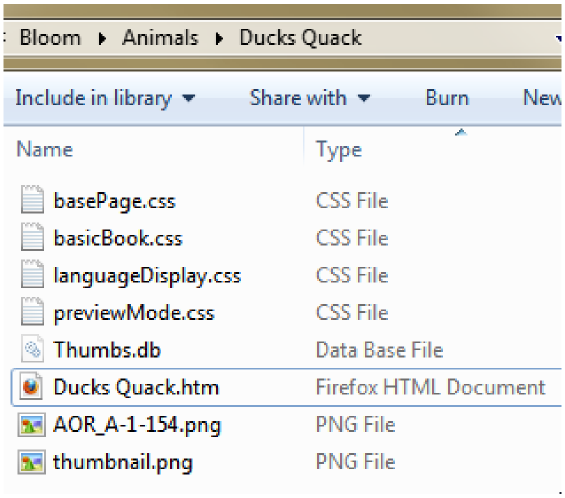


Ducks Quack
Frogs Croak
Animals.bloomLibrary

If you've created a Source Collection yourself, Bloom will place it in My Documents/Bloom. If the collection came with the Bloom installer, it's off in some "Program Files" directory. But if you've added the collection via a BloomPack, it will be in your ProgramData directory, as in "C:\programdata\sil\bloom\collections".

Books on disk

A book is actually a folder of files, stored inside a VC folder or Shell Book folder. Here's what the files look like:



Name	Type
basePage.css	CSS File
basicBook.css	CSS File
languageDisplay.css	CSS File
previewMode.css	CSS File
Thumbs.db	Data Base File
Ducks Quack.htm	Firefox HTML Document
AOR_A-1-154.png	PNG File
thumbnail.png	PNG File

This book has:

- An html document (.htm)
- A file for each illustration (.png)
- Several style sheet files (.css)[7] provide by Bloom
- Optionally, you can place a "book.css" style sheet here in order to override the others if customization is needed

If you ever have cause to move or copy a book from one Shell Collection or Vernacular Collection to another, you can do that by moving or copying one of these book folders.

Notice that the .htm document says "Firefox HTML Document"? Isn't Firefox a web browser? Yes, exactly!

> *In order to ensure that your investment in Bloom books is future safe, Bloom uses a restricted subset of the one of the most common, standardized formats on Earth: the web page.*

If you're thinking about commissioning new templates or front/back matter, you may be able to get by using the Template Maker, which is available if you're in a Source Collections. Until that is mature, you'll probably need to find someone with html/css skills (or someone willing to learn them). Information on the format is here. If you want something *really* fancy, like Bloom's Wall Calendar, which is smart enough to figure out the days of the year, you'll need someone with javascript skills. Information on building these *dynamic templates* is here.

[7] You'll note that there seem to be a lot of style sheets listed for just one document. Bloom applies different sheets to, for example, control what is editable during the edit process vs. when previewing the document. These are copied into the document folder only so that even on a machine with no Bloom, or if Bloom ever goes away, the document can still be opened in a Web Browser. Note that if you get a new version of Bloom, it will aggressively over-write the contents of these style sheets in each book that is opened.