Biblioblog Problems and Solutions:

PastoralEpistles.com as a Sandbox

R.W. Brannan

Logos Bible Software[1]

Version: 2005-11-08

ABSTRACT

Biblioblogs have come to serve a valuable role in the academic community: they disseminate information throughout the academy while serving to humanize their authors. But the information produced by bibliobloggers, while searchable via Google and able to be displayed upon request, is locked in an environment that stores content as a sort of lowest common denominator. Salient bits of information, ranging from discussions of particular Biblical passages to impromptu book reviews, are unable to be easily retrieved unless one happens to fortuitously stumble upon them. Most biblioblog content is first-rate, but the blogging software that typically generates and archives the content could be better.

This paper further defines some of these "lowest common denominator" problems and explores methodology used by PastoralEpistles.com to address them.

---

[1] Author email: **pe pastoralepistles com**, with **@** and **.** substituted for the spaces, respectively.

INTRODUCTION

A blog is the equivalent of a continually updated journal made available to the public on the internet. Software written specifically for this task exists in several forms,[2] allowing anyone with an internet connection and interest to compose blog entries in a matter of minutes.

Bloggers who focus on areas in the realm of biblical studies are known as "bibliobloggers". Announcements of new blogs from academics and others participating in the realm of biblical studies (called "biblioblogs") are now common. These biblioblogs are increasingly specialized in subject matter. Instead of focusing on (for example) the New Testament as a subject area, specific areas of study within the larger discipline are chosen.[3]

I currently operate two blogs and contribute to a third:

- **Personal Blog:** ricoblog (http://www.supakoo.com/rick/ricoblog)
- **Single Topic Blog:** PastoralEpistles.com (http://www.pastoralepistles.com)
- **Group Blog:** Logos Bible Software Blog (http://blog.logos.com)

I am currently employed by Logos Bible Software[4] as an Information Architect. My primary role at Logos involves the development Greek corpora within the Logos Bible Software environment.

THE SITUATION

In early 2005, I realized that a not insignificant proportion of my substantive posts to **ricoblog** dealt with the Pastoral Epistles. So I began to think about a separate blog focused on the Pastoral Epistles. But the sorts of things that I wanted to do in such a blog were not supported by the blogging software that I was currently using.[5] So it was time to experiment on all fronts: blogging on a focused topic and writing my own blog software that would explore how to address some of these problems.

THE PROBLEMS

Biblioblogging has become an effective mode for communication and development of ideas within the academic community. The lack of formality and the general acceptance of the biblioblog as a forum in which to think and write without the rigor of publication allows the biblioblogger to freely explore topics of interest.

---

[2] Blogger, MoveableType and TypePad are examples of blogging software combined with site hosting. Some are free, others charge a fee. James Tauber's Leonoardo is an example of blogging software offered outside of a site hosting service. See bibliography for URLs.

[3] Bibliobloggers seem to be moving from sites with wide designated areas of commenting (e.g. *NT Gateway Weblog, PaleoJudaica.com*) to specific areas of comment at times peppered with personal notes: *dokeo kago grapho soi kratistos theopholis* (Lukan writings), *Ralph the Sacred River* (Aramaic), and, of course, *PastoralEpistles.com*.

[4] Hereafter simply "Logos" unless context requires further specificity.

[5] The blogging software used by **ricoblog** is a package called *dasBlog*.

Many biblioblog posts contain valuable information. The unstructured approach to storing blog post content has the effect of locking data away from those who may find the information useful. This primary problem—the loss of ability to retrieve specific information that was adequately conveyed in its initial context—is exhibited in at least four ways:

1. The inability to reliably retrieve biblioblog posts that cite particular ancient documents, such as the Hebrew Bible, the Greek New Testament or the Apostolic Fathers.

2. The inability to reliably retrieve biblioblog posts that deal with particular subjects that may or may not be explicitly mentioned in the body of the blog post.

3. The inability to differentiate posts that have different natures—for example, simple link posts and longer posts on particular topics.

4. The inability to store bibliographic data in a meaningful and retrievable manner.

In short, blogs focus on content first and are concerned with structure secondarily. Because content is created and stored according to the lowest common denominator, future usage (particularly when attempting to aggregate data regarding particular books or subjects) becomes more difficult. Stefano Mazzocchi summarizes this clearly:

> … we all know that a complete mess is not a very good way to find stuff, so "data first" has to imply "structure later" to be able to achieve any useful capacity to manage information.[6]

If blogging systems begin to address underlying structural problems while keeping an emphasis on ease of data entry, biblioblog readers will be able to utilize the valuable information provided by bibliobloggers to a greater degree.

PROBLEM 1: POST CONTENT

Blogs are blunt instruments. They are functional and helpful. They utilize technology to convey information quickly and efficiently to readers. But the information is conveyed in a one-size-fits-all method. Consider the primary parts of the typical blog post:

- Post title
- Post author (unless a single-author blog, perhaps)
- Post categories
- Post date
- Post content

The post content itself, while arguably the most important component of the blog entry, is just a lump of text distinctively formatted in some way. It conveys the information that the author desires, but once the information is off of the biblioblog's front page or the biblioblog reader's feed aggregator,[7] how is the content ever found or accessed again?

There are a few ways, but none of them are really satisfactory:

---

[6] Mazzocchi, Stefano, "Data First vs. Structure First," n.p. [cited July 29, 2005]. Online: http://www.betaversion.org/~stefano/linotype/news/93/.

[7] Stand-alone applications designed to display RSS and/or ATOM structured feeds, such as SharpReader or BlogLines.

1. **Post Categories.** Most blogging software allows the assignment of categories to a given post. Category indices are available through links on the sidebar that call up a list of posts, perhaps sorted by some random criteria such as title or date, assigned to that particular category.

2. **Post Archives.** Several blogging solutions offer an interface to archives. This is typically a calendar interface is used to allow the blog reader selection of posts by day or month.

3. **Blog Searching.** Several blogs also support a blog-specific search. The site search box is found in the margin or sidebar of the blog front page. These searches are typically not super-capable; they often only offer simple string searches of blog content.

4. **Site-specific Google search.** Some bloggers have placed site-specific Google search boxes on their blogs to allow searching blogs in a familiar way. [8]

5. **Google Blog Search.** Google have recently introduced a blog-specific search feature[9] that allows searches to be tailored to search for particular topics on particular blogs.

6. **Google.** Google is perhaps the tool that best connects searchers with archived blog content.[10] Google indexes content as a matter of practice and due to its massive popularity is simply the largest source of new traffic around.

None of these options allow blog post content to be used any differently than other web content. They only allow page-linking and string-based searching. The most promising method—Google—treats blog content no differently than if it were any other content on the world wide web.

Biblioblogs contain specialized content that users may want to search in different ways. This information is helpful in the context of the post itself, but may be of more value when aggregated as an index. This sort of content includes but is not limited to:

1. **Words and Phrases in Ancient Languages.** Hebrew, Aramaic, Greek and Latin are common in Biblical studies. These languages are also common in biblioblogs.

2. **Biblical Citations.** Citations using Biblical references are common in biblioblogs. Some bibliobloggers actually go to the trouble to link these citations to online editions of the Bible.[11]

3. **Citations of Classical Documents.** Some biblioblogs cite corpora such as the Dead Sea Scrolls, the *Works of Philo*, the *Works of Josephus*, Aramaic Targums or the Apostolic Fathers.

---

[8] PastoralEpistles.com utilizes a site-specific Google search to allow readers to search site content. Another site that does this is Rubén Gómez' *Bible Software Review Weblog*.

[9] "Google Blog Search" n.p. [cited October 22, 2005] Online: http://blogsearch.google.com.

[10] A sample: At 9:51 PM PDT on June 29, 2005, **ricoblog** referral logs (via Site Meter) report that of the last 20 visitors, 12 were referred by Google. This percentage of hits originating from Google is not out of line with typical **ricoblog** referrals.

[11] e.g. Bible Gateway, zhubert.com, or the ESV web service.

Book readers utilize this sort of content through consulting indexes for particular words or citations and then examining the pages in question. Biblioblog readers would benefit from similar ability.

In a biblioblog this information is accessible through happenstance (*e.g.*, clicking the right category link and happening upon the right post) or through fortuitous string searching (*e.g.*, searching for something like "Mt 5.12" and finding an exact string match). Methods to reliably find and utilize this content are not available.

PROBLEM 2: POST METADATA

On the whole, blog posts contain the following metadata components:

- Post title
- Post author (perhaps)
- Post categories
- Post date

Consider a blog post that discusses a subject like Greek grammar. The blog itself may have a category of "grammar" or perhaps even "Greek grammar" under which the post will be filed. However, in a post on Greek grammar, one may discuss authors such as A.T. Robertson or Daniel Wallace. This hypothetical blog post may even cite examples of some esoteric point of grammar from the New Testament or perhaps allude to related topics.

Blogs have no method for capturing data such as these—topic[12] and reference data—even though it would be most helpful to later readers. Perhaps the reader has arrived as a result of a Google search and has no previous knowledge of the author's discussion on the topic, and would like to know more. This sort of information is available in printed books to the reader who chooses to examine tables of contents or indexes, but biblioblogs hide the data in plain sight.

This material should be available, in some form, to the biblioblog reader. It need not be displayed in every post during its time on the blog front page, but it should be accessible through indexes or through searches of blog content.

PROBLEM 3: POST TYPES

A third problem is the blog post itself. Even though authors compose different types of posts in the context of the same blog, the posts themselves are all the same internally. Some popular post types are:

- **Article posts.** This is the typical blog article. The definition of "typical" is different from blog to blog.

- **Link posts.** These are posts that have the primary function of pointing the reader to a different blog or web site. These include a link as well as a short explanation of relevance or comment of some sort.

---

[12] There are possibilities with the integration of services like Technorati and Delicious; whether these services provide the necessary solution remains to be seen.

- **Extended post.** Every blog author at times writes extended posts on topics of importance to the author. These are typically multiple paragraphs and, if noticed and thought relevant by other bibliobloggers, will become topics of discussion in the biblioblogosphere.[13]

- **Book review post.** Some blog authors use their blogs as a forum to offer short (or long) reviews of books they've recently read.

Of course, several other blog post styles exist, many of them specific to particular blogs. Jim Davila's "media watch" posts[14] may be considered a particular post type. An example of a book review or article review post type would be Torrey Seland's frequent and helpful posts announcing books or articles dealing with Philo of Alexandria.[15]

Blogs today have no such differentiation between posts. Each post has the same innate metadata. But perhaps different post types should have metadata specific to post type.

For example, perhaps the "link post" type could have, as metadata, the URL of the post being linked to and the name of the page (or blog) of the post being linked to. Perhaps a rating (on a scale of 1-10?) would also be appropriate in this context.[16]

The "extended post" type could perhaps list the blogs (and specific URLs) that end up linking to the extended article in question.[17]

The "book review post" type could list citation information such as author, title and ISBN for the book being reviewed, and include a link to the book's page on Amazon.com.[18] Linking to a published review in *Review of Biblical Literature* would also be helpful.

Distinction of this sort is not currently supported in blogging software. All blog posts are treated in exactly the same way. Appending different types of metadata to different classes of posts will provide for enhanced functionality, particularly among biblioblogs.

PROBLEM 4: BIBLIOGRAPHIES

Books are another focus of biblioblogs. Bibliobloggers tend to discuss books they are currently reading, cite from books they have read, or interact with other bibliobloggers about aspects of books.[19] But there are no easy methods beyond simple string searches for author or title to locate

---

[13] Some of this functionality is covered by "trackbacks", which allow blog authors to post referrals at the primary blog article to direct potential readers to the associated discussion.

[14] Davila, Jim. "Oxyrhynchus Papyri Watch." n.p. [cited June 5, 2005] Online: http://paleojudaica.blogspot.com/2005_05_01_paleojudaica_archive.html#111683945591209979.

[15] From Torrey Seland's *Philo of Alexandria Blog.*

[16] This is a simple example; social bookmarking web applications such as del.ico.us and Furl have largely filled this need.

[17] Again, trackbacks may be proper to use here. See note 12 above.

[18] This sort of functionality, relying on Amazon Web Services (AWS), was in the original specification for *PastoralEpistles.com* but has not yet been developed.

[19] A good example would be the dialogue between Edward Cook (*Ralph the Sacred River*), Jim Davila (*PaleoJudaica.com*) and Mark Goodacre (*NT Gateway Weblog*) about Aramaic retroversion from the Greek in *Aramaic*

discussion of certain books. Most bibliobloggers are diligent in linking to blogs that discuss common postings; but biblioblogs do not typically offer an easy way to search or browse post entries that deal specifically with particular books.

Providing browseable bibliographies is currently within the realm of possibility. A diligent biblioblogger could assign a category name such as "bibliographies", make sure to include a formal bibliographic citation within the post, and perhaps even name the post with the title of the book. But this is a lot of work for the biblioblogger, and it requires one to recall all of these somewhat arbitrary formatting conventions each and every time a bibliography entry is posted.

Even then, if all of this overhead is done with consistency and regularity, there is no way to search or sort by title or author. The keystrokes representing the bibliographical information are no different than the surrounding sea of letters that make up the balance of the post.

Existing blogging conventions leave valuable information locked away in posts. If a biblioblog discusses books that are available and relevant to the topic of the blog, chances are that other bibliobloggers and a portion of biblioblog readers would benefit from easy access to browseable bibliographies or indexes of this sort of content. But this specific capability simply is not available today.

THE SOLUTIONS

Before considering the implementations pursued, it must be noted that the software that runs PastoralEpistles.com is a prototype. The two-fold purpose is to post information about the Pastoral Epistles, and to think through and experiment with different features as applied to blogging. The resultant software that runs PastoralEpistles.com is called *Tychicus*.[20]

SOLUTION 1: ALLOW POSTS TO DISTINGUISH DIFFERENT TYPES OF CONTENT

One problem in the realm of content held within biblioblog posts is the inability to distinguish the specialized content from other post content. Potential topics, terms, ancient languages or citations are not distinguished apart from perhaps being linked to online databases or specification of different font information for different languages.

Tychicus is different. Information of this nature can be inserted at the time of post composition. Simple tags can be inserted around different types of content to distinguish text from surrounding content. These tags may or may not imply particular formatting.

To understand this at the proper level of detail, it is necessary to discuss post composition in Tychicus.

Sources of Mark's Gospel by Maurice Casey. Cook's first post (which started the dialogue) is online at: http://ralphriver.blogspot.com/2005/06/aramaic-wordplay-in-matt-39-par.html (accessed: June 29, 2005).

[20] Based on Eph 6.21 and Col 4.7, and Tychicus' appearance in 2Ti 4.12 and Titus 3.12.

COMPOSING POST CONTENT IN TYCHICUS

Blogs are only one form of internet publishing that has exploded in these first few years of the 21st century. Another form is the wiki.[21] Wikis are server-based content management systems that allow virtually anyone to create, edit or link to content that already exists in a given wiki. One of the hallmarks of wiki systems is simplicity. They focus on making common tasks easy. This is especially evident in entry of content.

Most wiki software supports a very simplified form of data entry. Wikis do not require users to directly use HTML tags. Most do not support WYSIWYG editing. Instead, very simple metacharacters are used as tags to represent formatting commands. A few examples are below:

```
= Heading Level 1

== Heading Level 2

This is normal paragraph text. This is /italic/ text. This is _underlined_ text.
This is *bold* text.

* bulleted list item
* bulleted list item
** indented bulleted list item

0 numbered list item 1
0 numbered list item 2
00 indented numbered list item
```

This style of structuring text is perfect for Tychicus.[22] It is easier for most to key than HTML tags and it allows the same basic capability. But most importantly, it is *extensible*. For Tychicus, the extended syntax involves the use of curved braces with a label to identify the sort of content contained in the tagged segment. Consider the below examples:

```
Here is some Greek text: {greek|*E)N A)RXH=| H)=N O( LO/GOJ}. Here is a link to
the ESV: {esv|1Ti 2.4-6}. Apostolic Fathers content can be linked to as well:
{af|EpDiog 5.1-3}. I can even specify a Google Search: {g|Pastoral Epistles}. Or a
link to a definition in Merriam-Webster's online dictionary: {mw|syntagmatic}.
```

This system is versatile. It allows a text string to serve as a label that informs the processor how to handle the text that follows. As is evident, Greek text and other languages are easily isolated. Different types of references are easily isolated as well. This information can be used by separate processes to generate indices of contents based on the tag label.

ADDING TOPICS

Supporting new tags is as simple as conceiving of a new label and adding a line or two of code to do the proper substitution. Adding capability to support the entry of data like author topics is really as simple as conceiving a new tag and adding support to render it.[23]

```
Authors to consult on the topic of {topic|verbal aspect} include {person|Stanley
E. Porter} and {person|Buist M. Fanning}. While Porter and Fanning do not agree on
every point, both authors are valuable to read and understand.
```

---

[21] The most well-known wiki is probably the Wikipedia, http://www.wikipedia.org.

[22] I am indebted to my good friend and long-time colleague at Logos, Eli Evans, for this idea and the basic conversion code that supports this feature.

[23] Generating documents that serve as indices to these tags is, of course, another matter. This is discussed in SOLUTION 2 below.

This sort of structure distinguishes the desired topics from other text in the post body with minimal tagging burden imposed on the author.

ADDING REFERENCES

Tagging references is done similarly. A label is created to denote the sort of reference included so that the reference information can be dealt with later:
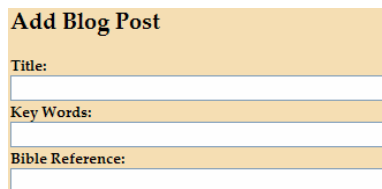
```
In {esv|1Ti 1.3}, Paul exhorts Timothy to "not teach any different doctrine".
Similar language is used by Polycarp in {af|Poly 3.1}: "Do not allow those who
appear trustworthy yet _who deliver contrary teachings_ daze you".[24]
```

The `esv` label indicates a Bible reference that is to link to an online edition of the English Standard Version. The `af` label indicates a reference to the writings of the Apostolic Fathers that is to link to an online edition.[25] These sorts of tags are converted into the proper style of reference when the post content is converted to HTML for display; the user is able to click on them to view the citation and retrieve further context if desired.

SOLUTION 2: ALLOW POSTS TO SPECIFY ARBITRARY METADATA

The first solution detailed a method to explicitly denote strings within post content as references or topics of some sort. The aforementioned solution is adequate, but it does not account for situations where a post does not explicitly state the topic or reference. There may be situations where ancillary or related information should be associated with the post, but the post content itself does not mention or cite the desired reference. In cases such as these, the ability to associate key words, topics or citations with a given post is necessary.

Tychicus supports this need by allowing simple strings of content to be associated with a post at the time of post composition. Separate input boxes for key words and Bible references are generated for each post type, according to the generic post metadata:

FIGURE 1

Key words are, by convention, delimited with either a comma or semicolon. References are also delimited in standard fashion, with commas delimiting verse references that share a common book and chapter (e.g. 1Ti 2.4-6, 8) and semicolons delimiting major chapter or book breaks (e.g. 1Ti 2.2-4; 3.15; 2Ti 2.22).

This information is available to later processes for generation of key word and reference indices. Because the key word or reference string has been absolutely associated with an article (or

---

[24] Poly *Phil.* 3.1 (Ehrman, 1:337).

[25] The *Apostolic Fathers Lookup Tool*. See bibliography for URL.

bibliography entry, or blog post), an index page may be generated from this information, serving as a reference to users interested in posts dealing with particular topics or to associated references.

DISPLAYING INDEX INFORMATION IN TYCHICUS

The practices of distinguishing different types of content in the post (e.g. the `{topic|…}` string denotation) and of specifying arbitrary metadata associated with the post combine to provide data that may be used to generate an index-style listing of post subjects and references.

Tychicus currently generates several types of indexes:

- Topic Index
- Person Index
- Place Index[26]
- Greek Index
- Latin Index
- Bible Reference Index
- Apostolic Fathers Reference Index

TOPIC INDEX

A "Topic Index" is a listing of references to articles that contain the specified topic as a key word.

Angels
> Question on B-Greek about 1Ti 3.16 and "seen of angels" in Blog Posts

Apostolic Fathers
> Reviews at RBL: Early Christians in Ephesus in Blog Posts
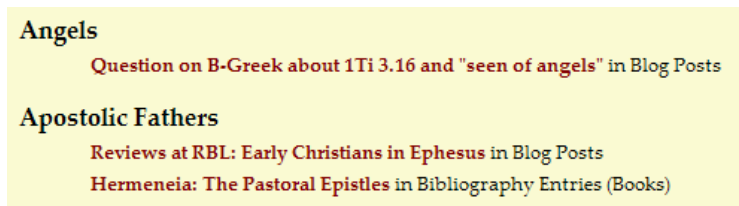> Hermeneia: The Pastoral Epistles in Bibliography Entries (Books)

FIGURE 2

On PastoralEpistles.com, this sort of information is gathered from multiple sources. The sources for topic/keyword index generation include:

- Topics specified at article entry in the article entry form.
- Topics specified at article entry in the article text (e.g. `{topic| … }`)
- Article categories specified at article entry.

To generate the topic index, a script is invoked. This script examines all post content, compiling indexes as XML files. The topics displayed above ("Angels" and "Apostolic Fathers") have XML as follows:

```
<term word="angels">
        <post type="blog-post" id="2005-05-04T02-35-39Z" title="Question on B-
Greek about 1Ti 3.16 and &quot;seen of angels&quot;" occur="1"/>
</term>
<term word="apostolic fathers">
        <post type="blog-post" id="2005-02-08T00-33-22Z" title="Reviews at RBL:
Early Christians in Ephesus" occur="1"/>
        <post type="biblio-book" id="2005-02-06T07-20-48Z" title="Hermeneia: The
```

---

[26] While supported, no posts currently specify information that would be compiled into a Place Index. I've included it in this list for completeness' sake.

```
          Pastoral Epistles" occur="2"/>
      </term>
```

The XML contains all information necessary to create the index: the post type, the post title, the post identifier (which is needed for the link) and even the number of occurrences of the topic in the article.

The script that generates the XML examines all post content for instances of the specified tag. All known item constructors are cycled, with each post for each item type examined for the specified tag. If an instance of the tag is located, the tag content is inserted into an XML node. After all post content has been examined, the XML DOM is saved to disk as an XML file. This file is then later used as the basis to generate the index when a request is made by a user.

PERSON INDEX

Person Indexes are generated in much the same manner as the Topic Index. The same basic script does all of the work, only instead of examining content for the `{topic| … }` tag, it examines content for the `{person| … }` tag.

Additionally bibliography-style posts are examined for author names. If an author is specified for a book, journal or URL bibliography entry, then the author's name is included in the person index.

GREEK AND LATIN INDEXES

As Greek and Latin keystrokes are specified by tags (e.g. `{greek|PISTO\J O( LO/GOJ}` or `{latin|firmamentum}`) the indexing script is able to compile indexes of each of these simply by examining each tag instance. The entire content of the tag (e.g. πιστὸς ὁ λόγος) is considered to be the index entry.

REFERENCE INDEXES

Tychicus also generates indexes for references. Currently only Bible references and references to the writings of the Apostolic Fathers[27] are indexed. This may be expanded in the future to include other patristic literature or literature such as the *Works of Josephus* or the *Works of Philo*. As regards display, this is similar to a word-based index:

Ge 2-3

Text in a Whirlwind: A Critique of Four Exegetical Devices at 1 Timothy 2.9-15 in Bibliography Entries (Books)

Lu 24.33-34

Question on B-Greek about 1Ti 3.16 and "seen of angels" in Blog Posts

Ac 7.27

"Rejecting" in 1Ti 1.19 in Blog Posts

FIGURE 3

Reference index information is gathered from the following locations:

---

[27] Tychicus does not currently support references to the *Shepherd of Hermas* as the structure of citation scheme is different than the balance of the works included in the corpus of the Apostolic Fathers.

- Bible references and references to the writings of the Apostolic Fathers that occur within post content that have been tagged as references.

- Bible references from post metadata that have been explicitly associated with the post content.

Post content and metadata is searched for instances of such references. This data is compiled into an XML index. The XML schema is much the same as that of the word index, adding a sort key that is used to canonically sort the reference index.

PARSING REFERENCES

One problem with indexing references is the problem of ensuring consistency in reference structure and naming. Bible book names are cited in several ways. A valuable index consolidates all naming variants under a particular style, so "1Ti 2.11" and "1 Tim. 2:11" are both indexed the same way. Tychicus' reference indexing takes this into account and has support for multiple forms of both Bible references and references to the Apostolic Fathers.

The same is true for personal names and other topics. Some consolidation is necessary if both "Stanley Porter" and "Stanley E. Porter" are listed as valid person topics for the blog. Librarians have been wrestling with this problem for years. Apart from the requirement of a subject authority[28] of some sort, the problem will continue. Chances are any index of people or topics that is automatically generated from short text articles composed over time will exhibit these sorts of deficiencies unless the generated indices are continually reviewed and articles continually edited to prevent this sort of aberration.

EXTENSIBILITY OF INDEXES

The index generation functionality creates indexes of post content based on a specified tag. To support new types of indexes, one must create a new tag and then add that string to the list of word index types. If metadata beyond tagging within the article exists, (e.g. author names in bibliography posts) some additional code would need to be written to extract this post information at the time of indexing.

SOLUTION 3: ALLOW DIFFERENT TYPES OF POSTS

Blogging software assumes only one post style: the blog post. The blog post is a chunk of HTML associated with an article title, some user-assigned categories and perhaps an author. But the goals of PastoralEpistles.com required different types of posts—different sorts of chunks of HTML having different things associated with them.

One way to accomplish this is to allow alternate post types. The original Tychicus specification called for the following post types:

- Blog Post
- Article/Essay
- Bibliography Entry (Book)

---

[28] For example, the Library of Congress has the Program for Cooperative Cataloging (PCC), which administers the Subject Authority Cooperative Program (SACO).

- Bibliography Entry (Periodical/Serial)
- Bibliography Entry (URL)
- File[29]
- Book Excerpt[30]

This is all well and good, and the idea is sound: Treat different sorts of information differently. But how to do it?

The solution lied in creating *Item Constructors*.[31] These are XML files that contain information for each sort of post type. Below is an example of the "Blog Post" item constructor:

```
<?xml version="1.0"?>
<tychicus-item-constructor type="blog-post" name="Blog Post">
        <metadata>
                <name>Blog Post</name>
                <description>Constructor for a blog post item.</description>
        </metadata>
        <form-info/>
        <display-preferences/>
        <bibliography-styles>
                <bibliography-style name="rico">
                        <element name="title" suffix="," html="i"/>
                        <element name="author" prefix="by" suffix="."/>
                </bibliography-style>
        </bibliography-styles>
</tychicus-item-constructor>
```

In this example, the item constructor does not contain information such as post author, post title, categories, or anything of that nature. This is because metadata that is held common between all posts has been collected and is part of a general constructor.[32] As the blog post only requires the generic post metadata, no additional metadata need be specified.

Between the generic post metadata and the post-type-specific metadata, the information necessary to construct a post is available. The generic post metadata includes:

- **Post Title:** This is the title of the post, as displayed in the blog article and in the RSS feed.

- **Key Words:** These are strings of text, delimited by comma or semi-colon (not spaces) that can be used to later generate a key word index of posts.

- **Bible Reference:** This is a string that can be interpreted as a single Bible reference or a series of Bible references. The references are then used to generate a reference-to-post index.

---

[29] PDF, HTML, etc. Essentially a link reference.

[30] Readers of **ricoblog** know that I've been working on a book dealing with "word studies" in the Pastoral Epistles. That is, not just considering what a given word means in general, but examining what a specific instance of a word means in its immediate context. Because of this, an additional purpose of *PastoralEpistles.com* (not discussed above) is to be an experimentation ground for ideas for the book and also perhaps as a venue to release excerpts under development for public feedback.

[31] In my original specification, I called each post an "item". Therefore an "item constructor" contained information on how to construct an item. That vocabulary doesn't really work in light of the usage of the term "post type". In spite of this, the term "item constructor" has stuck. "Post type constructor" just doesn't sound as nice.

[32] See APPENDIX I for an XML listing of the general form metadata, particularly the `form-info` element. See APPENDIX II for the Book-style Bibliography item constructor.

- **Description:** This component is where the wiki-style article text of the blog entry is composed.
- **Categories:** This is a series of elements that is rendered as a checkbox group. It allows the specification of categories for the current post.

Each post, regardless of post type, includes this generic metadata. Item constructors may specify additional metadata elements. An example would be the book-level bibliography post style. The below extra form information describes input elements for five different pieces of information that book-level bibliography entries require:[33]

- **Book Title:** This is the book title, as it should appear in any bibliography entry.

- **Book Author:** This is the author of the book in logical order: *Firstname Lastname*.

- **Publisher:** This is the publisher of the book.

- **Publisher Location:** This is the location of the publisher. The structure of this information is variable; it could be "Wheaton, IL" or "New York" or "London". As is typical with bibliographic style, major cities typically only require the city name to be represented, but smaller cities require the city and state or perhaps city and country.

- **Year Published:** This is the year that the book under review was published.

The input boxes generated from the post-specific constructor information are rather generic looking, but they accomplish the task at hand:

FIGURE 4

If different information is contained in the item constructor `form-info` element, then different interface elements will generate. The data contained in these form elements is captured and converted to XML, looking something like the below:

```
<book-title>Text in a Whirlwind: A Critique of Four Exegetical Devices at 1
Timothy 2.9-15</book-title>
<book-author>J.M. Holmes</book-author>
<publisher>Sheffield Academic Press</publisher>
<publisher-location>Sheffield</publisher-location>
<year-published>2000</year-published>
```

---

[33] The form should also include an element to record the number of pages in the book, however this was inadvertently overlooked in the original specification. Correcting this problem will require revision of all book-level bibliography data entered to this point.

The rendering of this information can be adapted for each context in which it appears. For example, bibliography information, at the time of post display, is rendered in a special way:

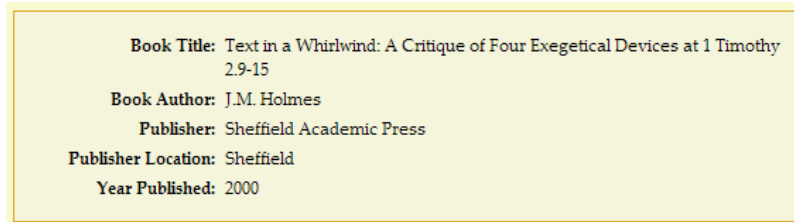|	
Book Title:	Text in a Whirlwind: A Critique of Four Exegetical Devices at 1 Timothy 2.9-15
Book Author:	J.M. Holmes
Publisher:	Sheffield Academic Press
Publisher Location:	Sheffield
Year Published:	2000

FIGURE 5

SOLUTION 4: ALLOW MULTIPLE BIBLIOGRAPHY STYLES

Once basic bibliographic data is stored and the post is differentiated as a bibliography, one must deal with the problem of rendering bibliography entries based on the stored data. The block-style component depicted in FIGURE 5 above is a generic solution that currently handles any extra `form-info` data, despite the post type. But this generic approach does not solve the issue raised in PROBLEM 4. A listing of all bibliography post types is generated,[34] but this does not serve as a bibliography listing as each post contains additional comments, quotations and other material regarding the book under discussion.

What is needed is a bibliography view in the traditional format that readers and browsers are used to: one line of text representing the primary information that describes the book in question. This information is now available due to its collection at the point of post entry, but the problem of display remains.

The solution to this problem involved updating the item constructor for each post type. A section detailing different bibliography styles is necessary. The below excerpt is from the book-style bibliography item constructor:[35]

```
<bibliography-styles>

    <bibliography-style name="rico">
        <element name="book-author" suffix="."/>
        <element name="book-title" suffix="." html="i"/>
        <element name="publisher-location" suffix=":"/>
        <element name="publisher" suffix=","/>
        <element name="year-published" suffix="."/>
    </bibliography-style>

    <bibliography-style name="title">
        <element name="book-title" html="i"/>
        <element name="book-author" prefix="by" suffix="."/>
        <element name="publisher-location" suffix=":"/>
        <element name="publisher" suffix=","/>
        <element name="year-published" suffix="."/>
    </bibliography-style>

</bibliography-styles>
```

Multiple bibliography styles are supported. Each style declares a name. Each style lists the required elements in the order required by the citation style. Additional information such as

---

[34] e.g. http://www.pastoralepistles.com/DisplayPostType.asp?type=biblio-book

[35] See APPENDIX II for a full listing of the book-style bibliography item component.

prefix data, suffix data and HTML rendering information is supported with attributes. The `name` attribute is not arbitrary, it refers to the element of the same name in the metadata supported by the current post type. Thus, the first `bibliography-style` group is specifying the following instructions:

1. Retrieve the text from the `book-author` element from the current item's metadata.[36]
2. Append "." to the `book-author`.
3. Retrieve the text from the `book-title` element from the current item's metadata.
4. Append "." to the `book-title`.
5. Surround the `book-title` in an HTML `<i>` element.
6. Retrieve the text from the `publisher-location` element from the current item's metadata.
7. Append ":" to the `publisher-location`.
8. Retrieve the text from the `publisher` element from the current item's metadata.
9. Append "," to the `publisher`.
10. Retrieve the text from the `year-published` element from the current item's metadata.
11. Append "." to the `year-published`.
12. Concatenate each element into a string in the above order, using a space as a delimiter.

Upon completing this process, a bibliography of the following style is displayed:

Holmes, J.M. *Text in a Whirlwind: A Critique of Four Exegetical Devices at 1 Timothy 2.9-15.* Sheffield: Sheffield Academic Press, 2000. [more detail]

FIGURE 6

Each item constructor is therefore able to specify multiple bibliography styles for a given post type. Each item constructor must specify at least one bibliography style. Because each post type supports a bibliography view, further functionality is supported:

- Post-type bibliographies[37] of multiple styles.
- A "site map" that serves a cumulative index of all posts, separated by post type.[38]

This style of display could be used in other scenarios. For instance, instead of listing a post title in a search result, a bibliographic citation could be listed instead. Bibliography-style entries can be used any time a post needs to be described in a short and meaningful way.

CONCLUSION

In this paper I've identified four areas where biblioblogs, as implemented with currently available technology, are lacking. These areas are:

Problem 1: Post Content Lacks Richness
Problem 2: Post Metadata Lack Richness
Problem 3: Posts Are "One Size Fits All"
Problem 4: Posts Lack Bibliographical Information

---

[36] `book-author` is stored *lastname, firstname*.

[37] e.g. http://www.pastoralepistles.com/DisplayIndex.asp?type=biblio-book

[38] e.g. http://www.pastoralepistles.com/SiteMap.asp

In addition to identifying these problematic areas, I've also discussed how my single-topic blog, PastoralEpistles.com, approaches these situations.

> **Solution 1:** Allow Posts to Distinguish Different Types of Content.
> **Solution 2:** Allow Posts to Specify Arbitrary Metadata
> **Solution 3:** Allow Different Types of Posts
> **Solution 4:** Allow Multiple Bibliography Styles

The primary problem is that currently available blogging software appeals first to ease of content creation. To encourage more people to blog (and thus using blogging software) writing blog posts must be easy. This is to be expected, the proliferation of blogs has been in part due to the simplicity of creating a blog and writing articles. However, in order for specialty-area blogs to provide more accessible content, they must offer more capability than simple online diaries.

In addition to allowing one to compose and post in a one-size-fits-all context, blogging software needs to allow specialists to *be* specialists. If there are common works (e.g. Bible text) blogging software should support use or citation of these common works. If there are common practices (e.g. bibliographical citations) then blogging software should take these practices into account.

The problem, of course, is the trade-off. Rich data requires more thought at time of entry, or it requires later editing. It also implies a user interface that supports and perhaps even encourages data enrichment. Instead of focusing only on entry of formatted text and the internal mechanics of posting an entry, blogging software needs to begin to account for structure and tagging to allow for the richer encoding of data.[39] Views, indexes and tools to compellingly exploit the richer data must also be developed.

This paper has addressed these issues in the context of a single, topic-focused blog. Other blogging styles exist and some of these suggestions would be useful outside of the topic-focused blog. Avenues of future exploration could include:

- The use of services like Delicious or Technorati as blog-specific topic indexes.

- Some blogging software (*e.g.* MoveableType) support plug-ins. Biblioblogging-centric plug-ins could be created that address topic and reference citation and indexes.

The advent of blogging, as applied to Biblical scholarship, has been a significant advance and allowed much more interaction in public discussion amongst the academy than has previously been available.

The next step in this progression could be specialized software allowing members of the academy to write and cite to the degree they see fit. This will provide archives of discussion searchable by various methods: by topic, by name, by language or by reference. And this step will further allow the content created by bibliobloggers to be both available and accessible to those who regularly read their blogs and those who casually happen across them in the context of internet searches.

---

[39] Mazzocchi, Stefano, "Data First vs. Structure First," n.p. [cited July 29, 2005]. Online: http://www.betaversion.org/~stefano/linotype/news/93/.

BIBLIOGRAPHY

This bibliography is broken up into sections, grouping relevant links together for ease of reference. When a web site in general (such as a blog) is referenced, only the author, title and URL are given. When specific articles are cited, a complete citation is provided.

BLOGGING SOFTWARE

Blogger/BlogSpot. (http://www.blogger.com)

dasBlog (http://wiki.shahine.com/default.aspx/DasBlog.HomePage)

Leonardo. (http://jtauber.com/leonardo)

MoveableType. (http://www.moveabletype.com)

TypePad. (http://www.typepad.com)

BOOKS

Apostolic Fathers. 2003. Translated by Bart Ehrman. 2 vols. LCL. Cambridge: Harvard University Press.

Casey, Maurice. *Aramaic Sources of Mark's Gospel.* Cambridge: Cambridge University Press, 1998.

Holy Bible, English Standard Version. Wheaton: Crossway Books, 2002.

FEED READERS

BlogLines (http://www.bloglines.com)

SharpReader (http://www.sharpreader.net)

WEB SITES AND ONLINE ARTICLES

ARTICLES

Brannan, R.W. "Apostolic Fathers Lookup Tool." No pages. Cited October 1, 2005. Online: http://www.supakoo.com/rick/af.asp.

——— "Convert Greek Beta Code to Normalized Unicode." No pages. Cited October 1, 2005. Online: http://www.supakoo.com/rick/ConvertGreek.asp.

Carter, Joe. "Notes on Blogging: The 5/150 Principle." No pages. Cited September 8, 2005. Online: http://www.evangelicaloutpost.com/archives/001584.html.

Cook, Edward. "Aramaic Wordplay in Matt. 3:9 par.?" No pages. Cited June 29, 2005. Online: http://ralphriver.blogspot.com/2005/06/aramaic-wordplay-in-matt-39-par.html.

Davila, Jim. "Oxyrhynchus Papyri Watch." No pages. Cited June 5, 2005. Online: http://paleojudaica.blogspot.com/2005_05_01_paleojudaica_archive.html#111683945591209979.

Farrell, Henry. "The Blogosphere as a Carnival of Ideas." No pages. Cited November 8, 2005. Online: http://chronicle.com/free/v52/i07/07b01401.htm

Mazzocchi, Stefano. "Data First vs. Structure First." No pages. Cited July 29, 2005. Online: http://www.betaversion.org/~stefano/linotype/news/93/.

Sifry, Dave. "State of the Blogosphere, August 2005, Part 3: Tags." No pages. Cited August 6, 2005. Online: http://www.technorati.com/weblog/2005/08/37.html.

Tauber, James K. "The World Wide Web as an Academic Forum: A Case Study of the Hellenistic Greek Linguistics Pages." No pages. Cited August 11, 2005. Online: http://ausweb.scu.edu.au/aw95/education4/tauber/.

BIBLIOBLOGS

See *Biblioblogs.com* for an ever-growing list of biblioblogs.

Anderson, Richard. *dokeo kago grapho soi kratistos theopholis* (http://kratistostheophilos.blogspot.com)

Brannan, R.W. *ricoblog* (http://www.supakoo.com/rick/ricoblog)

——— *PastoralEpistles.com* (http://www.pastoralepistles.com)

Carlson, Stephen C. *Hypotyposeis* (http://www.hypotyposeis.org/weblog)

Cook, Ed. *Ralph the Sacred River* (http://ralphriver.blogspot.com)

Davila, Jim. *PaleoJudaica.com* (http://paleojudaica.blogspot.com)

Goodacre, Mark. *NT Gateway Weblog* (http://www.ntgateway.com/weblog)

Rosson, Loren, III. *The Busybody* (http://lorenrosson.blogspot.com)

Seland, Torrey. *Philo of Alexandria Weblog* (http://philoblogger.blogspot.com)

OTHER BLOGS

Gomez, Ruben. *Bible Software Review Weblog* (http://www.bsreview.org/weblog)

Logos Bible Software. *Logos Bible Software Blog* (http://blog.logos.com)

Tauber, James K. *James Tauber's Blog.* (http://jtauber.com/blog)

OTHER WEB SITES

Amazon Web Services (http://www.amazon.com/gp/aws/landing.html)

BiBIL Biblical Bibliography of Lausanne: Books of the New Testament (https://wwwdbunil.unil.ch/bibil/?MIval=/bi/en/bibilhome&BiMenu=10&BiMain=6&vvedid=4248)

Bible Gateway (http://www.biblegateway.com)

del.icio.us (http://del.icio.us)

ESV Bible Web Service (http://www.gnpcb.org/esv/share/services/)

Furl (http://www.furl.net)

Google (http://www.google.com)

Google Blog Search (http://blogsearch.google.com)

Hubert, Zack. *zhubert.com* (http://www.zhubert.com)

Logos Bible Software (http://www.logos.com)

Program for Cooperative Cataloging (http://www.loc.gov/catdir/pcc/)

SiteMeter (http://www.sitemeter.com)

Society of Biblical Literature. *Review of Biblical Literature* (http://www.bookreviews.org)

Subject Authority Cooperative Program (http://www.loc.gov/catdir/pcc/saco/saco.html)

Technorati (http://www.technorati.com)

W3C XML Specification (http://www.w3.org/TR/2004/REC-xml-20040204/)

Wikipedia (http://www.wikipedia.org)

APPENDIX I: GENERAL FORM METADATA

The general form metadata is available in a common XML file. Form elements allowing entry of this information is always provided for every type of post supported by Tychicus.

```
<!-- post title -->
<element type="input" style="text" label="Title" name="iptTitle" />

<!-- keywords -->
<element type="input" style="text" label="Key Words" name="iptKeyWords" />

<!-- Bible ref as index -->
<element type="input" style="text" label="Bible Reference"
name="iptBibleReference" />

<!-- public/private -->
<element type="input" style="checkbox" label="Private" name="cbPrivate" />

<!-- textarea for wiki-style entry -->
<element type="textarea" style="short" label="Description" name="txtDescription"
/>

<!-- checkbox group for categories -->
<element type="checkbox-group" label="Categories" name="cbCategories">
        <checkbox name="first-timothy" value="First Timothy">First
                Timothy</checkbox>
        <checkbox name="second-timothy" value="Second Timothy">Second
                Timothy</checkbox>
        <checkbox name="titus" value="Titus">Titus</checkbox>
        <checkbox name="greek" value="Greek">Greek</checkbox>
        <checkbox name="latin" value="Latin">Latin</checkbox>
        <checkbox name="apostolic-fathers" value="Apostolic
                Fathers">Apostolic Fathers</checkbox>
        <checkbox name="cross-references" value="Cross-References">Cross-
                References</checkbox>
        <checkbox name="septuagint" value="Septuagint">Septuagint
                </checkbox>
        <checkbox name="authorship" value="Authorship">Authorship
                </checkbox>
        <checkbox name="grammar" value="Grammar">Grammar</checkbox>
        <checkbox name="morphology" value="Morphology">Morphology
                </checkbox>
        <checkbox name="syntax" value="Syntax">Syntax</checkbox>
        <checkbox name="exegesis" value="Exegesis">Exegesis</checkbox>
        <checkbox name="sermons" value="Sermons">Sermons</checkbox>
        <checkbox name="commentaries" value="Commentaries">Commentaries
                </checkbox>
        <checkbox name="lexical-studies" value="Lexical Studies">Lexical
                Studies</checkbox>
        <checkbox name="background-studies" value="Background Studies">
                Background Studies</checkbox>
        <checkbox name="textual-criticism" value="Textual Criticism">
                Textual Criticism</checkbox>
        <checkbox name="miscellaneous" value="Miscellaneous">Miscellaneous
                </checkbox>
        <checkbox name="about-this-site" value="About This Site">About
                This Site</checkbox>
</element>
```

APPENDIX II: BOOK-STYLE BIBLIOGRAPHY ITEM CONSTRUCTOR

```xml
<?xml version="1.0"?>
<tychicus-item-constructor type="biblio-book" name="Bibliography Entry (Book)">
        <metadata>
                <name>Bibliography (Book)</name>
                <description>Constructor for a bibliography item, classifying a
book.</description>
        </metadata>

        <form-info>
                <element type="input" style="text" label="Book Title"
                        name="iptBookTitle"/>
                <element type="input" style="text" label="Book Author"
                        name="iptBookAuthor"/>
                <element type="input" style="text" label="Publisher"
                        name="iptPublisher"/>
                <element type="input" style="text" label="Publisher
                        Location" name="iptPublisherLocation"/>
                <element type="input" style="text" label="Year Published"
                        name="iptYearPublished"/>
        </form-info>

        <display-preferences/>

        <bibliography-styles>

                <bibliography-style name="rico">
                        <element name="book-author" suffix="."/>
                        <element name="book-title" suffix="." html="i"/>
                        <element name="publisher-location" suffix=":"/>
                        <element name="publisher" suffix=","/>
                        <element name="year-published" suffix="."/>
                </bibliography-style>

                <bibliography-style name="title">
                        <element name="book-title" html="i"/>
                        <element name="book-author" prefix="by" suffix="."/>
                        <element name="publisher-location" suffix=":"/>
                        <element name="publisher" suffix=","/>
                        <element name="year-published" suffix="."/>
                </bibliography-style>

                <bibliography-style name="publisher">
                        <element name="publisher" suffix=":"/>
                        <element name="publisher-location" suffix=","/>
                        <element name="year-published" suffix="."/>
                        <element name="book-title" html="i"/>
                        <element name="book-author" prefix="by" suffix="."/>
                </bibliography-style>

                <bibliography-style name="year">
                        <element name="year-published" suffix="."/>
                        <element name="book-title" html="i"/>
                        <element name="book-author" prefix="by" suffix="."/>
                        <element name="publisher-location" suffix=":"/>
                        <element name="publisher" suffix="."/>
                </bibliography-style>

        </bibliography-styles>
</tychicus-item-constructor>
```