BHL Full Text Search Planning

Version 0.2 – November 28 2016

This document includes the specifications and user interface for the BHL Full Text Search. It also contains specifications on how the ElasticSearch indexes will be set up.

This document is not editable, but you may add comments which will be incorporated into the document at a later date.

Requirements

Main Search Box

The main search box of BHL will send the search to ElasticSearch (ES) and will default to searching books, journals and articles and will be labeled **Publications**. And the search term will be applied to metadata and full text. The relevance ranking of search results will be such that the metadata is ranked higher than the full text. And some metadata fields, such as title or subtitle will be ranked higher than the others.

Search terms that match on Title and Subtitle (or similar) are ranked highest, other Metadata are ranked second highest, then OCR Text.

Items are returned by the search. Not Titles. This means that a search term that appears in multiple volumes of the same series will return things that look very similar. The volume number will need to be part of the title of the data that is returned.

TODO: Identify and specify what fields to use and what rankings they should have.

Tabs

Tabs across the top will remain, with the following tabs displayed:

- Publications
- Authors
- Subjects

- Scientific Names

Each tab will have their own set of results, but only publications will have facets. All tabs will use ElasticSearch to standardize on functionality. The existing tabs of “Books/Journals” and “Articles/Chapters/Treatments” will be merged into the new **Publications** tab.

The remaining three tabs, **Authors**, **Subjects** and **Scientific Names**, will provide search results identical to what we currently have, a simple list of results. ES provides a potentially better fuzzy matching algorithm for misspelled names, which might make for better search results on these tabs.

Each of the tabs will show the number of results for that particular search, in the same manner that we currently have on the site.

The index will be updated daily with updates and additions made that day.

Questions:

- Is it possible or feasible to update the index at the same time that the item is being saved? Or is it preferable to wait until the middle of the night (U.S. time) to update the index for those items that changed in the past 24 hours? Would it make sense to queue up the items that need to be updated in the index and then update it more often during the day?

- Will the counts of results for each of the tabs be limited to the first 100 or 500 results? (It seems like that is what happens now.)

Advanced Search

Advanced search will also submit to ES for search results for Books and Articles, but not for others. The advanced search will build a more customized ES search that provides additional fields to the search to “pre-narrow” the results. We should implement advanced search in ES only if it’s relatively simple to construct a more detailed search to ES.

TODO: Look at the advanced search more carefully. Create a diagram of what the new advanced search might look like.

Pagination

Breaking the results up into pages will help with page load times and is a familiar feature for users. This will also help preserve bandwidth (a minor concern at the source, but possibly more of a concern for non-US users) and render times for users with slower computers.

Allow the user to choose how many results per page that they want to see. (25, 50, 100, 250). That should be enough. Default to 25. More than 250 per page is probably not needed.

Facets

Fields to facet on are (in this order?)

- Type (Book, Journal, Field Books, Article, etc)
- Author
- Date of Publication
- Contributor
- Subject
- Language

Metadata Concerns

The data in the facets on are deeply rooted in the metadata we have and therefore we will likely uncover lots of instances of strange or “bad” metadata.

Type Metadata

Type metadata consists of the bibliographic level data from the Genre field combined with the Material Type data. When constructing the data to include as the facet, the Genre (book, journal) should be used unless there is a Material Type (field book, visual work, and others if they are added).

Place Metadata

We will not have a facet for Place of Publication. We will revisit the idea of Place as facet as a phase two activity. This would include analyzing the OCR content to identify place names in the text of a book and then distill those down into metadata that could be used as facets.

Scientific Name Metadata

We will not have a facet for Scientific Name. There are potentially too many results when

searching across all items in BHL. Instead the scientific name facet will appear when searching within a book.

As a phase two (or three) effort, we will investigate and consider adding Genus to the search results in an effort to possibly narrow the number of items displayed in this facet.

Date Metadata

Date facets will be displayed in more granular fashion than the years that we display in the Browse by Date button at the top of the homepage. The date ranges and item (not segment) counts are:

- 1450-1580 (136 items)
- 1581-1699 (564 items)
- 1700-1799 (3039 items)
- 1800-1824 (approx. 3200 items)
- 1825-1849 (approx. 4200 items)
- 1850-1874 (approx. 10000 items)
- 1875-1899 (approx. 18800 items)
- 1900-1924 (43110 items)
- 1925-1949 (6290 items)
- 1950-1974 (6722 items)
- 1975-1999 (7089 items)
- 2000-2016 (1847 items)

We will aim to use the existing logic to convert the dates that we have into these facets for the ES Search results. This will provide consistency to the end user even if the date translation isn't perfect. This is also something we can continue to refine as users test and respond with feedback.

Sorting

In this phase, there will be no ability to sort the results. It's expected that the facets will be more useful to select which results the user may be interested in. We may revisit sorting after we receive feedback during testing, but the addition of sorting means additional work and possibly more complicated indexing.

Within-book searching

When viewing a book, a new search box will be included to search only the OCR text of that book. This search will not use any metadata and will focus solely on providing matching terms in the text of the item.

The results will include links to the page of the book that contain the search term along with a text snippet with the search term highlighted. The results will also include facets for **Page Type** and **Scientific Name**.

If possible, the search results should be displayed in such a way as to be able to switch between the results and the normal page-viewer without redoing the search. That is, displaying the search results in a element that can be shown or hidden via Javascript.

The final specs for this are (Jan 9, 2018):

- Add to the Book Viewer a Search tab that mirrors the OCR and Annotations tabs' functionality. This allows search results to be presented side-by-side with the page images.
- Search hits will be highlighted in the result snippets. Hits within the OCR and page images will not be highlighted at this time, but may be considered for a future enhancement.
- Search results will be ordered by page/leaf, and not by any type of ranking value (like “number of hits”).
- Facets will not be included at this time. Since this includes page types, the "type" of pages that appear in search results will be shown (i.e. "Page 142 (Illustration)"). Facets may be considered for a future enhancement.
- While the user is waiting for search results to be presented, a status indicator should be displayed.

Other Features

Download Results and View more Books/Journals... buttons

Current website has these two buttons. Do we want to retain these? How much is **Download** used? If we retain it, it needs to bypass pagination, but we may still want to limit the number

of results just for performance reasons (500 entries? 1000?).

With pagination, then **View More** becomes redundant.

Stopwords

It may be wise to investigate the list of stopwords used by ES and decide whether there are words that we want to exclude from the index that are biodiversity-specific but don't provide relevant or useful search results. (Low priority, nice to have)

Milestones

No dates on these yet. *Still in progress...*

- ElasticSearch experimentation (Mike)
- Selection and Loading of Sample Data (Joel)
- Initial Alpha Version Testing with Sample Data (Mike, Tech Team, Selected Staff)
- Refine or respond to Alpha Version feedback (Mike)
- Installation and Configuration of Server (Joel)
- All content indexed and ES part of weekly ingest (Mike)
- Beta release to BHL Staff (Mike/Joel)
- Launch Date (Mike)

Implementation

Reconfigure BHL to get OCR text from the ES Server
Ingest Process pushes OCR Content to the ES Server

Currently the OCR data is stored on a location on the Smithsonian network that is less than ideal in terms of performance when accessing all of the OCR data for indexing purposes. The new search server will have enough space to store all of the OCR data (with room to grow) which will be redundant, backed up, and faster to index since it will be on the same server and not over the network.

To switch to this, Mike indicates that this is a simple matter of pointing the App server to a different network location for the files. I'm assuming that the Ingest process will also respond to this change and few changes need to be made there.

Ingest Process indexes new content into ES
Metadata Updates cause item to be reindexed in ES

This requires a change to the IA Ingest procedures and to the item save process in the BHL Admin Portal. Requires more analysis. *Still in progress...*

Code Changes for the tabs to search ES instead of the current code

Tabs remain, but the URLs they call will need to be redone to interface with the new ES search code. Requires more analysis. *Still in progress...*

IU Changes for searching within a book

Add the search box on the book view page.

Add a search results to the book view page with links that jump to that page in the book (without a page reload, if possible?)

Allow switching between the book reader and the search results, because this seems like it would be a common feature.

Other UI changes

The bibliographic info page will need to be modified to accept a volume identifier (or item identifier) from the URL in order to pre-select the proper volume of a series. This functionality does not yet exist in BHL. It will need to be added.

Reporting for ES performance and statistics

Would it be nice to know how the search is being used? How many bits are indexed. We will aim to be sure to configure Google Analytics to (continue to) record the search terms that users are entering.

Index Specification

INDEX 1: Books and segments

- Includes the full text of each book/segment.
- Enables searching for books/segments, and presenting results in a single list.

- Enables finding a book/segment that includes a search term in the text.
- Enables faceting books/segments by a variety of fields.
- Facets
    - Type (Book, Journal, Field Book)
    - Author
    - Place of Publication (needs to be normalized)
    - Date of Publication (needs to be normalized)
    - Scientific Name (here there be dragons)
    - Contributor
    - Language
    - Subject

INDEX 2: Pages

- **Includes the text of each page.**
- Names are attributes of pages.
- Enables drilling into a given book/segment to find pages matching search term.
- Enables searches for a name on any page in any book.
- Enables faceting pages by scientific name or page type.
- Search term highlight within the search results

INDEX 3: Authors

- Powers the Authors tab and advanced search
- Provides fuzzy match searching on Author names

INDEX 4: Subjects / Keywords

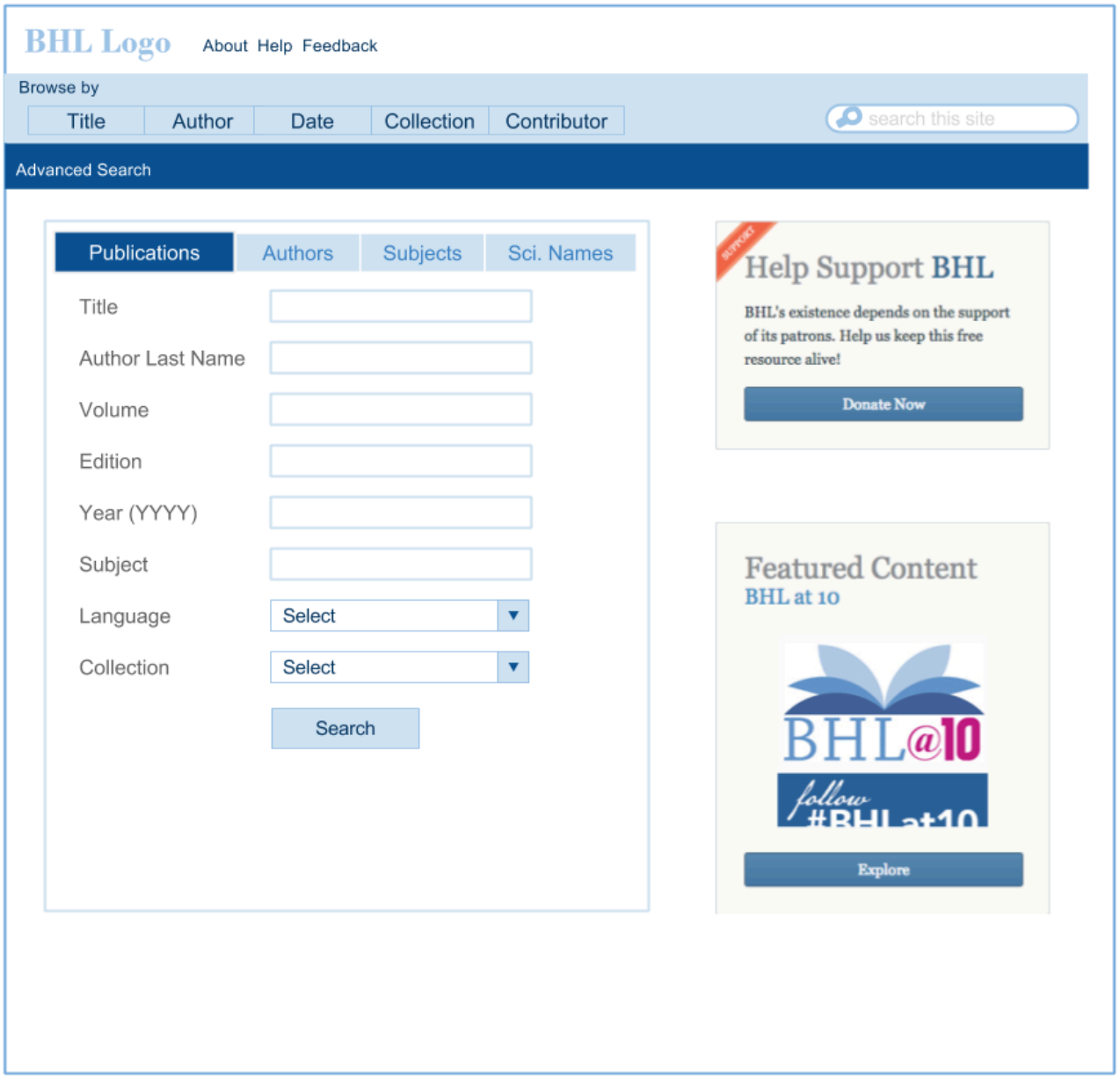
- Powers the Subjects / Keywords tab and advanced search
- Provides fuzzy match searching on Subject names
- May be able to display a histogram of books published over time with that subject (similar to Google's n-grams)

INDEX 5: Scientific Names

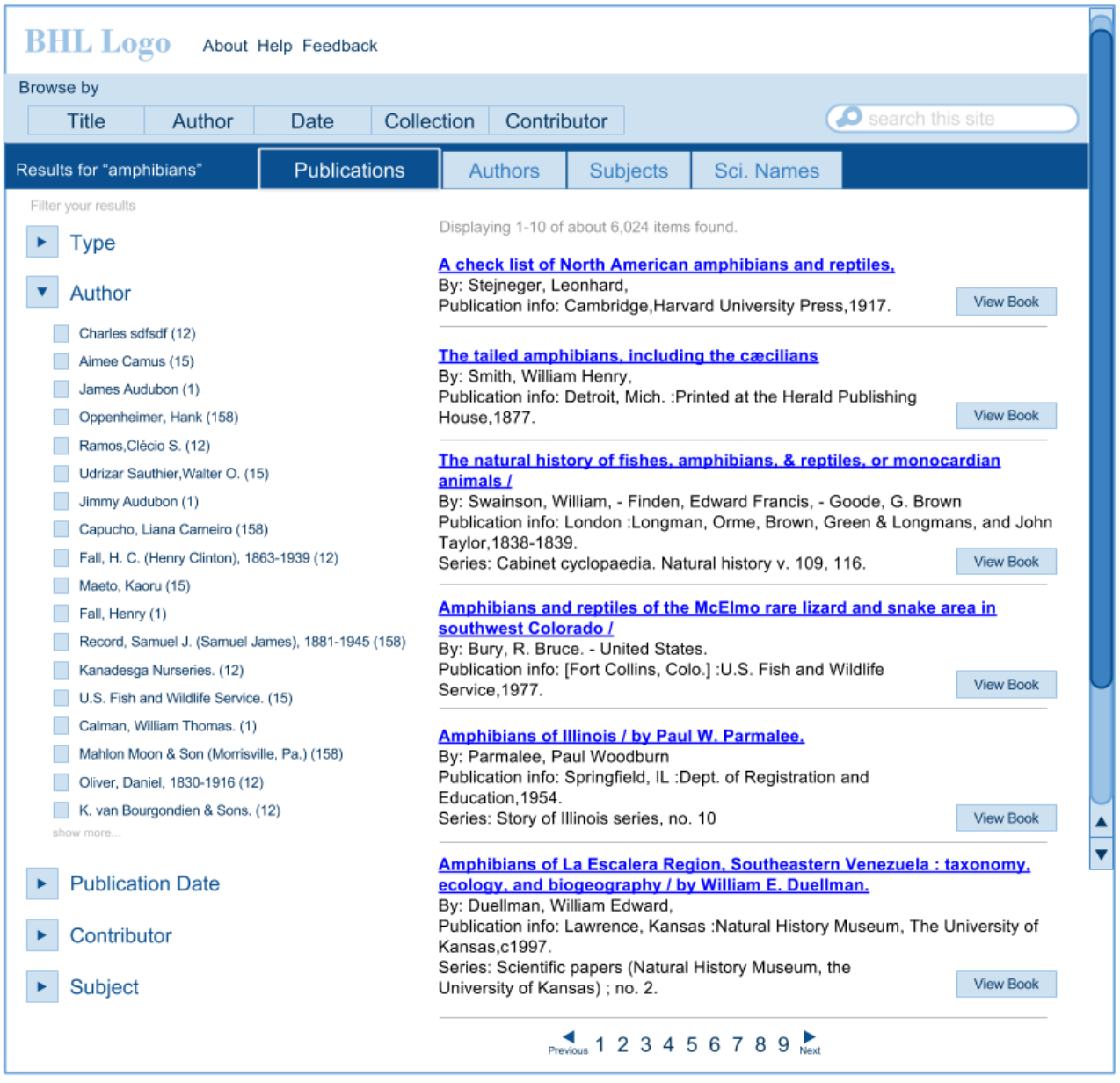
- Powers the Scientific Names tab and advanced search
- Provides fuzzy match searching on Scientific Names names

Diagrams

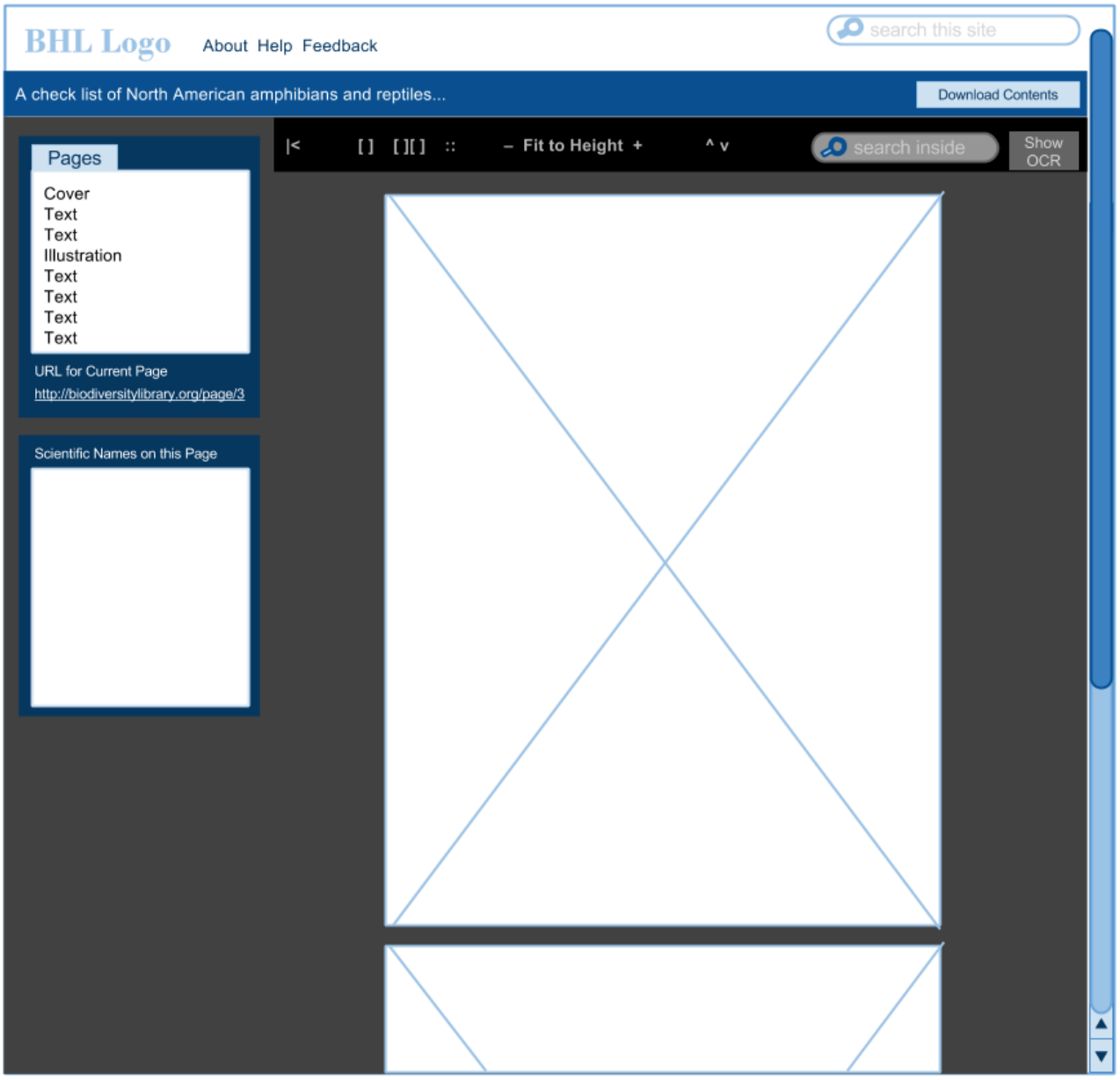
The Advanced Search doesn't change much with the exception that the tabs are combined into one **Publications** tab and that Title will search both Book and Journal titles as well as Article and Segment titles.



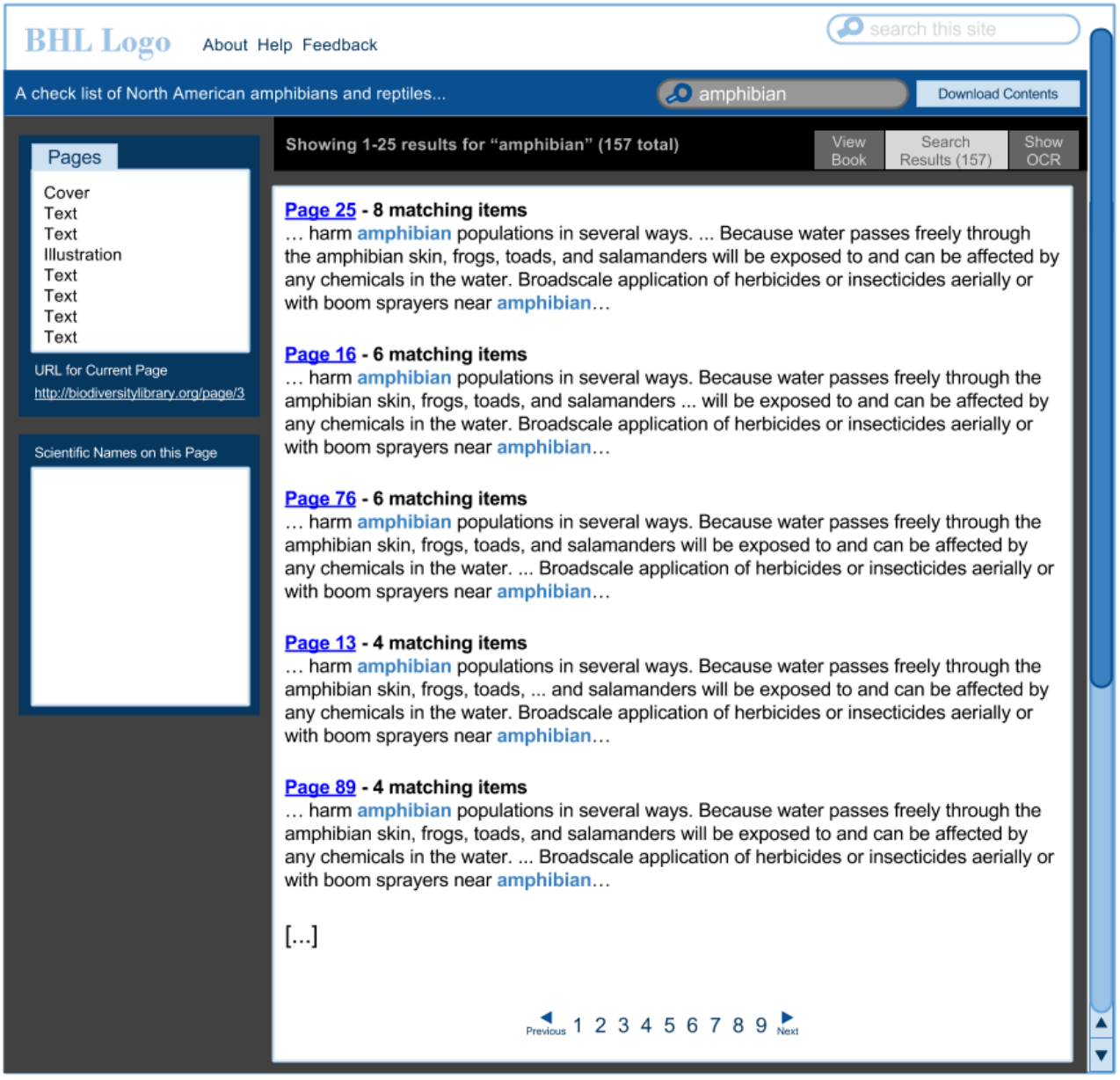
The **Search Results** look more or less the same as what we have now, with the removal of the *Contributed By* text removed. The facets appear on the left and are expandable for a cleaner interface. The **show more...** link causes the list of facets to expand downwards showing 15-20 (?) more facets each time it's clicked. The font size for the facets should be small but legible to fit as many facets as possible on the page.



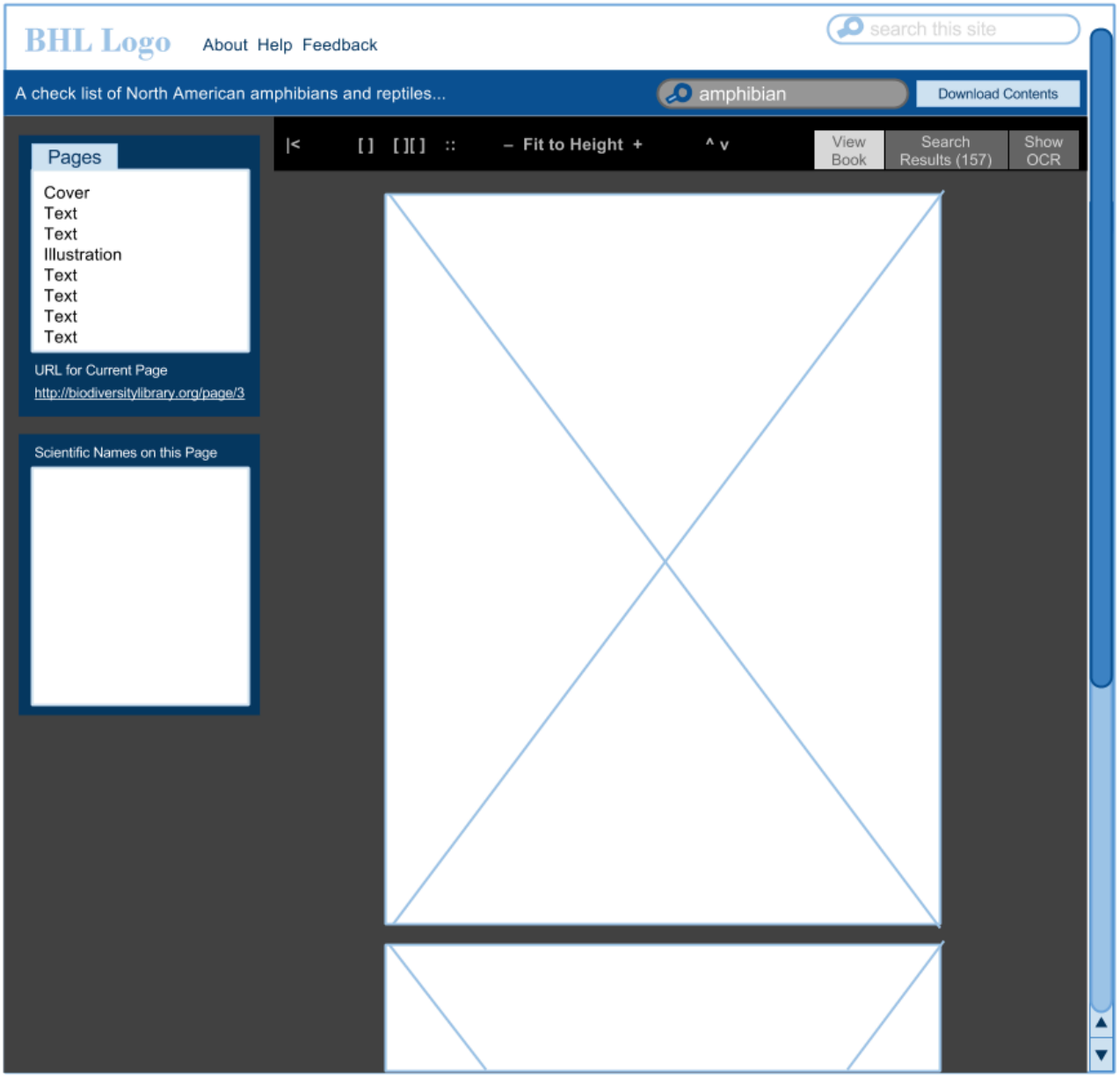
The View Book page is the normal layout with the addition of the “search inside” search box displayed alongside the Title and the Download Contents button. This should be enough to indicate that one can search *within* the current book.



The results for searching inside a book includes additional navigation tabs for switching between the search results and viewing the book. Visually the page is the same, indicating the user is still viewing a book. Clicking a result heading (the blue page number) activates the **View Book** tab and scrolls to that page. (This function should already exist). The **Show OCR** tab remains unchanged. There is no search results highlighting in the **Show OCR** view.



This view shows how the display looks when viewing a page while search results are present. The **View Book** is now activated and the **Search Results** is deactivated. Note that the **Search Results** tab includes the number of results. Switching between the two tabs should not redo the search but should rely on AJAX and Javascript for a cleaner user experience.



Server Specifications

The server that we have ordered to handle full text search will perform two duties:

1. Acting as the full text search service
2. Serving OCR files to the website

We are finding that performance is not ideal for accessing the OCR content for large operations, such as indexing. Therefore, to make this smoother, and place the OCR text as close as possible to ElasticSearch, the new server will have enough space to house the OCR text, with plenty of room to grow for the next five years.

Specifications

- Dell PowerEdge R730 Rack-mounted Server
- Intel Xeon E5-2630 v4 2.2GHz (10 cores)
- 128 GB RAM
- 4 x 4.0 TB Traditional Hard disk 7200 RPM
- Gigabit ethernet
- RAID 5 for about 12 TB of space on the Traditional HDDs
- Operating System: RedHat Enterprise Linux version 7
- Software: ElasitcSearch

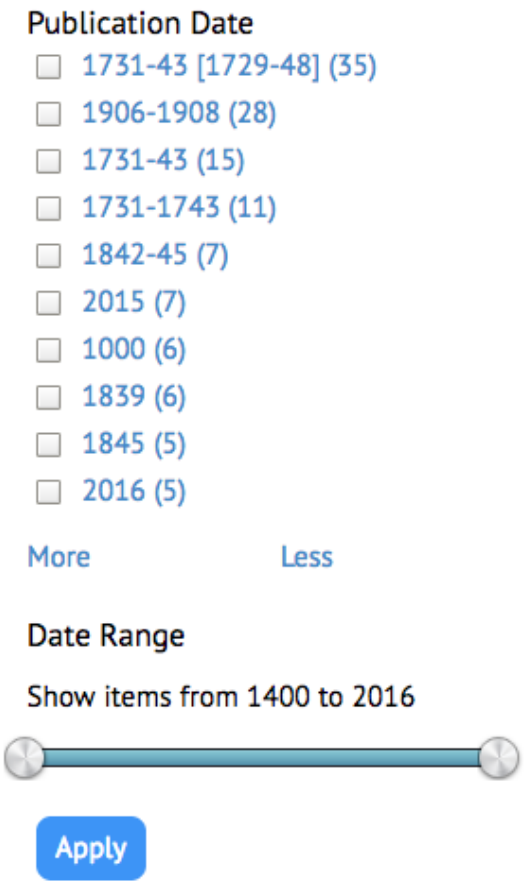
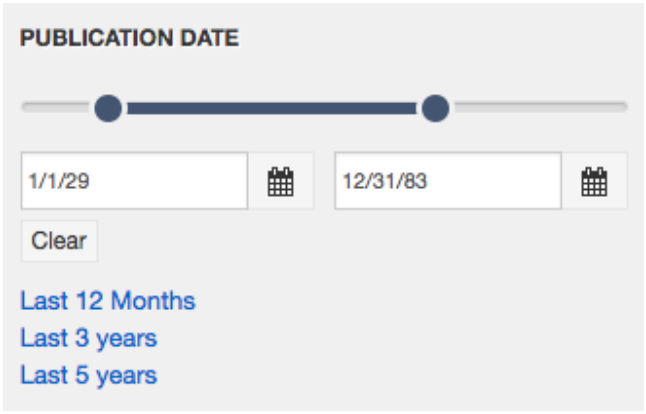
Collections for Alpha Testing

Use these BHL Collections to create the test data for items across BHL for testing in the Alpha version of the Full Text Search.

Art of Science	11,400 pages
BHL at 10	13,800 pages
BHL Field Notes Project	5,500 pages
Bone Wars Collection	7,700 pages
Carl Linnaeus Collection	7,800 pages
Charles Darwin's Library	174,000 pages
contents of the journal “Phytologia”	52,700 pages
Latino Natural History	34,100 pages
Top BHL Partner Content	14,000 pages
TOTAL	**About 321,000 pages**

This is sufficient enough in terms of numbers and should have a variety of content to test against. Additionally, the size of the index is small enough that reindexing should take only 20-30 minutes.

Appendix A: Date Facet Examples

Smithsonian Libraries Site Search	**Notes:**
	A combination of checkboxes and sliders. The checkboxes definitely have “odd” data. The slider is more normalized and possibly preferred.
Summon (SI Libraries OneSearch)	**Notes:**
	These search results are commonly used for more recent publications, hence the inclusion of “Last 12 months”, etc. They also use a date slider. Not sure how effective it is.

HathiTrust **Date of Publication** 1910-1919 (12,424) 1900-1909 (10,085) 1980-1989 (8,022) 1990-1999 (7,724) 1970-1979 (7,095) 1890-1899 (6,354) 1920-1929 (5,808) 2000-2009 (5,578) 1880-1889 (4,277) 1960-1969 (3,238) 1870-1879 (2,346) 1940-1949 (2,107) 1950-1959 (2,092) 1930-1939 (1,678) 1922 (1,593) 1914 (1,462) 1921 (1,413) 1912 (1,406)	**Notes:** This uses decades to break out the results, which means they must be processing and normalizing the data on the individual search results. These are also ranked by number of results.
WorldCat Year 2016 (5080) 2015 (6677) 2014 (5699) 2013 (4775) 2012 (5231) 2011 (5147) 2010 (5047) 2009 (4621) 2008 (4511) 2007 (4425) 2006 (3999) 2005 (3656) 2004 (3222) 2003 (3031) 2002 (2740)	**Notes:** Broken out by individual year, this makes it very difficult to jump down to something published many decades ago. Not the best interface. A “Show More” link (not displayed) is available, but it simply shows more years in order and takes several clicks to get to 1900.

Freer Gallery of Art	**Notes:**
Browse by › Object Type › Topic › Name › Place ⌄ Date + 1690s(112) + 1640s(93) + 1700s(82) + 1800s(81) + 1860s(81) + 1720s(76) + 1610s(75) + 1650s(75) + 1660s(75) + 1620s(73) + 1630s(73)	Also broken out by decade, sorted by number of results.

Appendix B: Notes on how ElasticSearch weights fields

When the a document (BHL item) is being indexed, three things are looked at in each field. A field may be the title, keywords, or full OCR text.

1. The number of times a term appears in the field
2. The length of the field
3. How often that term appears in all fields in documents.

If a field is short, like the title, then a term appearing in that field is weighted higher.

If a field is very long, and a term appears many times in the field, and many times over all of the text, then that term is weighted lower, like *a*, *and*, or *the*.

If a field is very long, and a term appears infrequently, then it's ranked higher. Like *hippopotamus* or *giraffe*.

All three of these are combined to rank the document in the search results. On top of this, we can force certain fields to be ranked a little higher than others such as our metadata to encourage the index to rank those terms higher, but changing the weights of the fields will cause us to need to reindex the entire corpus of data.

Nitty Gritty Details:
https://www.elastic.co/guide/en/elasticsearch/guide/current/scoring-theory.html