After bringing their card catalogs online in the 90s, many libraries have maintained decades-old interfaces even as they adopt new systems focused on the discovery and use of subscription materials such as single journal articles. Though antiquated in design, such "classic catalogs" remain popular with librarians and faculty. As libraries move into a third generation of catalogs and discovery systems, those charged with leading such transitions are choosing to rethink the classic catalog or do away with it entirely. This paper visits the history of library catalogs on the web, examining the necessary maintenance introduced by each generation. It concludes with an assessment of the difficult decisions around labor and maintenance faced by those hoping to improve access to their materials.

"the Cat lives! Long Live the Cat! a functional collection search engine! … I would rather use a paper card catalogue than a one-box search." (@TheMedievalDrK, June 14, 2018)

"Today was the first day that I noticed that a number of 'classic catalogues' have been taken offline by their universities. According to the error message generated by one of these universities, this move is a permanent one. (Which leads me to infer that the coincidental 'offline' status of two others is not-so-coincidental).

"Should anyone responsible for making such decisions be reading this email: I hope you come to appreciate the gravity of your error & work diligently to improve the copy-specific / print-format filters that are now required to overcome the digital bias of Discovery interfaces." - Jason Rovito

For many college/university librarians and faculty, the “classic catalog” remains their preferred method of accessing materials held by their library. These web interfaces, often developed and adopted in the 1990s, are maintained online even as libraries adopt new systems focusing on the discovery and use of subscription materials such as single journal articles. In this third generation of catalogs and online “discovery” systems, many library administrators are

---

1 Jason Rovito, email to EXLIBRIS-L listserv “RIP classic catalogues,” 2019-04-30. N.b. The EXLIBRIS listserv has no relation beyond wordplay to the Ex Libris library vendor mentioned elsewhere in this paper.

2 Library discovery systems combine indexes of materials from the library’s catalog with metadata records other sources, such as subscriptions to services which offer journal articles, newspaper articles, and more. These may include hundreds of millions of materials, compared to the approximately 5-12 million
choosing to commission new web interfaces to their classic catalogs or decommission them entirely.

This moment offers an opportunity to reflect on the history of library catalogs on the web, focusing on the labor of maintaining each system. In this paper, I will take the reader through the three generations of library catalog websites, both their basic functions and the labor and maintenance aspects of each generation’s developments, and end with an assessment of the difficult decisions of the current landscape. I will use the catalog and discovery systems at my own institution, the Penn State University Libraries, as an example throughout.

A Note on Underlying Systems

In order to understand the web presentation of library catalogs, we must understand the underlying software. The first Integrated Library Systems (ILSes) were developed in the late 1970s to manage library business functions in a shared system. Over time, they expanded from maintaining and sharing library catalog records to include modules for managing acquisition of and payment for materials, circulation of the materials, electronic resource access, holds, and more. Library workers perform tasks in these individual modules, generally using desktop clients, which all update a shared database.3

Because the ILS is central to library business functions and migrations cause library-wide disruptions for workers and patrons alike, many libraries still use systems well over a decade old and migrate rarely.4 For example, Penn State uses Sirsi’s Symphony system, which we adopted in its earlier form, Unicorn, in 1999. To put its age in context, Unicorn was first installed at Georgia Tech in 1982. Though introducing some updates, the move to Symphony did not require replacing all of our Unicorn documentation, most still applied. We perform our daily work in Symphony through a Java desktop application called Workflows. When we migrated to Sirsi Unicorn 20 years ago, the Workflows software we adopted was little different in core design than that used today.

Library workers maintain three copies of the ILS—production, QA, and development—on three locally-hosted servers. Each of these require both operating system and software upgrades, security patches, and common server maintenance.

________________________________________
4 Marshall Breeding’s annual library systems reports (previously “automation marketplace”) are fascinating reading, although they require some familiarity with companies and systems. They illuminate how many libraries still use older systems https://librarytechnology.org/industryreports/
Gen 0: In the Beginning, There Was Telnet

Before library catalog websites, ILSes provided an access port via telnet, an early command-line client-server protocol. Users with some kind of networked access could telnet to a public index of the library’s catalog and search for materials. Rather than use a physical card catalog, patrons could search their own library’s catalog from terminals in the building or from other locations, such as campus offices. But, more than that, anyone with network access could search any library catalog which had a public telnet interface. As ILSes added account management, patrons became able to perform rudimentary tasks via telnet, such as accessing a list of items they had checked out.

Although it did not have the same demands as a website, providing telnet access to an ILS created an additional workload. Library workers not only maintained the ILS on local servers, they managed the indexing of fields for telnet search, and configured access to the public telnet port.

*Systems maintained*: integrated library system (including QA), telnet index and settings, open web port

Gen 1: Before it was Classic

As more people acquired personal computers and internet access speeds improved, ILS vendors began to offer catalogs as web applications, built using languages like C, Java, COBOL, and Perl. These websites came out of the box with the ILS product and offered limited opportunities for customization. When Penn State Libraries purchased the SirsiDynix Unicorn (now Symphony) ILS in 1999, it included the web catalog application still available today as the classic catalog at https://cat.libraries.psu.edu.

In developing these catalogs, vendor and library developers focused on re-creating aspects of the card catalog experience. Default indexes prioritized title search and browse, author search and browse, and subject search and browse. Such indexes work very well for users seeking a known title or author or who are familiar with library cataloging practices and controlled terms.

Unfortunately, catalog web applications often reflect the practices of the early web. Penn State’s classic catalog, for example, is very difficult to use on mobile devices. It was also built in an era of session timeouts (generally after about 20 minutes of inactivity) and requires the user to navigate to previous pages by clicking navigation buttons within the page. Attempting to use the browser’s Back button causes an error. Its session-generated URLs cannot be bookmarked.

---

5 In the days before personal computers and dial-up home internet, this was often only available in networked settings such as in major academic or government centers.
Changing the display look and feel can only be done through editing small, opaquely-named html and CSS files which represent a minimal portion of the page.

*Systems maintained*: integrated library system (including QA and development servers), catalog web interface, catalog index, possibly telnet settings

**Gen 2: Classic Catalog in the Age of Discovery**

The modern academic library provides access to far more materials than can be found in its catalog. As more journal articles and scholarly materials became available online, libraries sought methods to provide unified search, rather than requiring users to visit subject-specific databases. Vendors responded by developing the first generation of what are known as “discovery” systems. These systems are typically integrated with the “Knowledge Base,” a system through which library workers manage the access to online materials available through library subscriptions. However, neither the Knowledge Base nor discovery interface are part of the Integrated Library System.

In 2012, Penn State Libraries adopted the Serials Solutions electronic resource management service and deployed its public interface, Summon. Summon attempts to emulate Google by providing a single box which searches all materials in the library’s catalog and to which the library has subscription access. Although Summon is hosted by its vendor, Ex Libris, it still requires some maintenance by library workers. Customization options are limited, but maintainers must ensure that error pages provide functional links, may change the text labels of various boxes, and can make minimal changes to look and feel. Maintainers must remain aware of system upgrades which may break current changes as well as changes on the library’s side which must be updated in customization.

The largest ongoing maintenance task results from Summon’s complete separation from the ILS. Library workers must provide Ex Libris with an up-to-date copy of the catalog. This is done using a locally-developed script which runs daily and identifies new, changed, or deleted records, extracts copies of said records, and uses FTP to place each kind of update in a designated directory where Summon’s system expects to find them. The script must also ignore a subset of catalog records which should not be sent to Summon, due to restrictions on access. Each day, Summon sends reports of any errors in changes received, which must be reviewed and corrected.

Although discovery systems had been promoted as a replacement for the traditional library catalog search, the user experience soon became one of information overload. As of May 2019, Penn State’s Summon index included records for 784 million objects. Of these, only a little over 7 million came from the library’s catalog.\(^\text{6}\) This, as well as the need for certain business

---

\(^\text{6}\) For reference, on May 6, 2019, the online catalog of the library of congress ([https://catalog.loc.gov/](https://catalog.loc.gov/)) described itself as having 17 million records, still 750 million fewer than Summon.
functions such as ILS account management (holds, renewals, etc.), led to the continued maintenance of the original web catalog, now called the classic catalog. Library workers now maintained two entirely separate systems and managed their interrelation.

**Systems maintained:** integrated library system (including QA and development), catalog web interface, catalog index, discovery system, discovery system knowledge base, catalog record updates in discovery system

**Gen 3: Catalogs in the Cloud**

The third generation of online catalogs promises either a great reintegration or a reimagining, depending on the route an institution chooses. Ex Libris’s Alma, with most of its adoption occurring in the past four years despite its 2012 release, integrates management of electronic resources into a traditional ILS. It bundles this with one of the two Ex Libris front-end discovery systems, providing users with what might be termed a truly integrated library system. Alma is only available as software-as-a-service, meaning that the subscription covers much of the cost of maintenance. Rather than require workers to login via desktop clients, it provides a web interface which any worker can log into anywhere. A few other vendors, such as Sirsi, are in the process of developing cloud-based interfaces to their library systems, but Alma remains what its marketing materials describe as “the only unified library services platform in the world” at the time of this writing.

Depending on the original ILS, the move to Alma may require years of planning and data migration. Workers must change decades-old workflows. Its integrated design eliminates the need to send daily updates to a separate discovery system, but other forms of maintenance must be identified and managed.

For those not choosing to migrate to Alma, what are we to do with classic catalogs whose designs and practices are older than most of our graduating class? Some continue to maintain them, while directing most library users to their second-generation discovery systems. At Penn State, we have begun the process of rebuilding the classic catalog using Blacklight, an open source Ruby application which uses a Solr index. We will emulate many of the data-driven features of our old catalog, while providing the permanent links to pages, 21st century user experience practices, responsive design, and similar behaviors one might expect of a website in 2019.

---

8 The thought of replacing an application such as Sirsi’s Workflows with a modern browser interface is tempting.
10 The most popular functional competitor to Blacklight is VuFind, which introduces a similar level of maintenance.
Our choice introduces a third system in need of maintenance. We made this trade-off because we did not identify sufficient benefits in Alma to outweigh the combined financial cost, time cost of migration, and the trauma of extended disruptions of nearly every library worker’s established processes. Instead, we will take on the cost of maintaining an additional system, with the intent of providing our patrons a better user experience.

Systems maintained, Blacklight/VuFind: integrated library system (including QA and development), locally-developed catalog web interface in Blacklight/VuFind, indexing application, index in Solr Cloud (replicated across several “shards”), discovery system, catalog record updates in discovery system

Systems maintained, Alma: Alma, web interface of discovery system, discovery system knowledge base

Death and Rebirth of the Classic Catalog

While Alma brings libraries back to the earlier ages of the truly integrated library services platform, it eliminates the public classic catalog. It is the first major ILS-type software to do so. The spate of recent migrations is likely the cause of the change referred to in the listserv email cited in this paper’s opener. For the majority of people whose user task is simply “search known item, find item, get item,” a well-implemented Alma instance may be an adequate substitute for the classic catalog. For library workers, maintenance is now focused on a single tool, albeit one with many facets which require review and management. Yet the classic catalog’s death is mourned by the skilled researcher, such as Dr. Kennedy in the opener, who must now use blunter tools.

The rebirth of the classic catalog we see at institutions like Penn State comes with a high cost of labor and maintenance. The software requires much customization to be ready for deployment. The opportunity to control all aspects of the look and feel, unlike in previous systems, means that workers must make decisions about all such aspects. Besides maintaining the first generation ILS and second generation discovery system, workers must maintain all the dependencies of a Ruby application and Solr index, balance server space against site traffic, and provide QA and development forms of the site as well. They must write indexing schemas, host indexing applications, and write scripts which extract data regularly from the ILS and perform indexing. In Penn State’s case, the production site adds around 6 new servers to our environment, although we will eventually take the classic catalog offline, reducing our maintenance burden somewhat.

Unlike vendor-developed products, there is no fallback to a service package or support ticket if things go wrong with these new catalog applications. One must rely on community support and perform the work oneself. Thus, only libraries who can afford a team of programmers can afford to make such choices. Libraries with smaller staff but sufficient budgets (or consortial purchasing opportunities) may be able to move to Alma with its high cost, onerous migration,
and loss of the classic catalog. Those who can do neither must continue as-is, maintaining a 20th century web application which becomes increasingly difficult to use.
Bibliography
