Applying a Software Development Product Cycle to Library Technology Adoption and Development

Rutgers University has made this article freely available. Please share how this access benefits you. Your story matters. [https://rucore.libraries.rutgers.edu/rutgers-lib/55890/story/]

This work is an ACCEPTED MANUSCRIPT (AM)
This is the author's manuscript for a work that has been accepted for publication. Changes resulting from the publishing process, such as copyediting, final layout, and pagination, may not be reflected in this document. The publisher takes permanent responsibility for the work. Content and layout follow publisher's submission requirements.

Citation for this version and the definitive version are shown below.

**Citation to Publisher Version:** Costello, Laura. (2018). Applying a Software Development Product Cycle to Library Technology Adoption and Development. *Journal of Library Administration* 58(4), 334-345. [https://doi.org/10.1080/01930826.2018.1448649].


Terms of Use: Copyright for scholarly resources published in RUcore is retained by the copyright holder. By virtue of its appearance in this open access medium, you are free to use this resource, with proper attribution, in educational and other non-commercial settings. Other uses, such as reproduction or republication, may require the permission of the copyright holder.

Article begins on next page
Title: Applying a Software Development Product Cycle to Library Technology Adoption and Development

Abstract: Product cycles have long been used in software development, but libraries can also benefit from a structured product life cycle to help manage technology development and acquisition. This article outlines a proposed product cycle for library technology management with information on implementing each step of the cycle. It will also include information for librarians assessing the success of structured technology management.

Keywords: software development, product cycle, emerging technologies, library technology, project management, technology, libraries

Laura Costello is the Virtual Reference Librarian at Rutgers University Libraries, Alexander Library, 169 College Ave., New Brunswick, NJ 08904. laura.costello@rutgers.edu

Received: October 26, 2017
Accepted: January 11, 2018
Applying a Software Development Product Cycle to Library Technology Adoption and Development

Library technologies are continuously developing, and technologies that might be relevant to libraries are increasing in a broad range of disciplines and fields. In this environment, it is essential to establish a clear protocol for the development or adoption of new technologies and for the maintenance and eventual replacement of these technologies. A product cycle, like those used in software development, can be a practical tool for libraries overwhelmed by the rapid emergence and obsolescence of technology trends and tools. A product cycle can help institutions evaluate their existing technologies, look for new solutions, and know when it is time for a change. An established product cycle can also help libraries establish the suitability of emerging technologies for their current and future needs.

Technology evaluation is not unique to libraries, and the process of technology evaluation is broadly applicable. As the work of librarians becomes more interdisciplinary, the range of technology products relevant to libraries is expanding. In academic libraries, this could mean technologies used by departments and scholars to find information in their fields. In public libraries, this could include every technology patrons use to find information or access content. While these expansions signify a need to diversify reference and programming in libraries, they also complicate the understanding of which technologies libraries should invest time and money providing to patrons. In order to establish the best use of limited funds and resources, an established methodology should be in place to help evaluate the suitability of these technologies for library problems and practice.

The evaluation of new practices and technologies is also a long-term problem that will continue to grow in importance. Library technologies will continue to grow and change, so establishing an iterative technology-evaluation process that is applicable to today’s technologies while opening the possibility of integrating new technologies is an essential practice for libraries of all kinds. This paper lays out a protocol to encourage mindful technology integration in
APPLYING SOFTWARE DEV PRODUCT CYCLE

libraries including evaluating new products, creating a plan for when technologies slowly or suddenly fail, and generating a productive response to technologies that no longer meet our needs.

**Literature Review**

The concept of the product cycle has roots in business and technology development. Philosophically, this idea grows out of the industrial quality-control efforts of the mid-20th century, particularly the work of W.A. Shewhart at Bell Labs as summarized in his book, *Economic Control of Quality of Manufactured Product* (1931). Shewhart traces the origins of his theories to the scientific method (pp.3-7) and applies them to quality-control efforts specific to physical product. Though Shewhart focused on products, his theories were one of many management structures applied to software development from its earliest origins (Umpleby & Dent, 1999).

Deming also adapted Shewhart’s theories into a cycle that has had a strong influence on software development (Yilmaz & Chatterjee, 1997). The Plan-Do-Study-Act (PDSA) Cycle (Deming, 1994) outlines an iterative process for product development and assessment that focuses on planning, research, and testing in the “Plan” phase; decision-making and product implementation in the “Do” phase; launch assessment in the “Study” phase; and action based on the study in the “Act” phase (pp.132). This type of iterative process, with a few adjustments, is a natural fit for software development and assessment. The cycle has influenced the growth of software development and splintered into several cyclical development processes, including the Agile Methodology, an iterative software development process that is currently popular (Bustard, 2012).

There are other management strategies that flow from the work of Shewhart and his contemporaries that have been successfully applied in software development or IT workflows. While this article focuses primarily on a cyclical product cycle growing out of Deming’s (1994) interpretation of the theories, Fox (2016) has written a comprehensive explanation of several
iterative development theories from software development and other corporate sectors and their potential applications in to projects and services in libraries. While Fox suggests that corporate strategies might not be directly applicable to the goals and methods of libraries, learning from each stage in an iterative process has the potential to make library initiatives more efficient. Mathews’ white paper (2012) applies ideas inspired by software development to the library environment, including advocating for a cyclical product-development cycle based on the Lean Startup methodology. The Lean methodology was developed by Eric Ries (n.d.) and focuses on iterating on the needs of early adopters to build a user base. DeRidder (2007) also describes a technology selection process that shares several concepts with this cyclical model, including assessing a potential problem or user group, assessing and researching possible technology tools that could serve that purpose, and testing. This article goes into great detail regarding cost assessment and has many practical recommendations that remain relevant.

A few examples of libraries using a version of a software-development product cycle can be drawn from the literature. One example is the development of an early online-discovery system at Cornell University Library. In their complex development process, they suggest an iterative development model that integrates feedback from stakeholders throughout the process (Calhoun, Koltay, & Weissman, 1999). University of California San Diego Libraries used an Agile-inspired development cycle to build their mobile website. The Agile inspiration was evident in their timeline of short development cycles followed by assessment, feedback, and integration back into the development cycle (Critchlow, Friedman, & Suchy, 2010). The Crawford County Federated Library System used a bespoke product-development cycle in creating their open-source authentication software, Libki. Their cycle included a testing period with library staff followed by assessment and integration of feedback and an additional cycle in which user feedback was collected and integrated before the system was upgraded (Hall, Ames, & Brice, 2013). The software-development cycles in these libraries share several important features with
the product cycle outlined in this article, including needs-based development, testing, and a maintenance process that loops back into development.

While the examples above represent development processes that might be used verbatim by libraries engaged in their own software development, libraries integrating and evaluating existing technologies can also use the framework of the product cycle to evaluate technologies and think critically about features and systems that they would like to see developed for their existing needs. An example of this is the assessment framework for discovery systems developed at Emory University. The framework is deliberately scalable beyond existing technologies (Durante & Wang, 2012). This type of critical technology assessment applies a process to acquiring new technologies while making room for the technologies that will eventually replace them. The product cycle can be used to develop this type of cycle in libraries that are developing their own technologies, those that are working with vendors to improve existing technologies, and those that only have the resources to acquire out-of-the-box technologies that have already been piloted in other institutions.

Applying the Product Cycle to Libraries

The product cycle outlined in this paper was developed through work as a non-technical project manager on library development teams and through curation and selection of technologies for libraries that do not develop their own software products. This cycle is applicable to both strategies and seeks to outline a process for identifying library needs, technology selection, and maintenance. The figure below shows a visual representation of the proposed product development cycle for libraries.

[Insert Figure 1]

Library technology life cycle differs from product cycles established for business in that library technology is largely not-for-profit and is developed or adopted to meet the needs of a defined group of patrons or a prospective group of library users. This article outlines a proposed model with suggestions for each step of the process. Individual libraries may adapt this model to
meet their specific workflows. Some libraries might find it useful to assign responsibilities for each step of the process to different group members or library departments, while in smaller institutions this process may be conducted by a single librarian. The cycle is not meant to be followed to the letter, but rather to function as an outline for managing the adoption or creation of a library technology.

Establishing the need from an existing problem is the first step in the proposed product cycle. Problem-centered methodologies might work best for libraries because limited budgets often restrict how widely libraries are able to experiment with new and emerging technologies. A product cycle that is initiated by the needs of the community served by the library ensures that an active user community already exists for new technology projects. Beginning the cycle with a need also allows project managers to approach software development and acquisition with a documented need in place, rather than trying to forecast need or to back-justify the integration of a technology.

Needs will sometimes be apparent. For example, a library might have a need to share data between departments, reach new patron audiences, or find better ways for patrons to search and access library resources. Needs can also take the form of a specific or general request from a patron for a product or service the library could provide. Needs can be established through a strategic plan or administrative directive. Because this process is cyclical, needs can also arise through problems with an existing technology or a technology that is not adapting to the growing needs of the library.

Project managers can also conduct needs assessments or general evaluations if needs are not apparent. A good example of a formal needs assessment is Ohio University's survey of its students about technologies they used and the technologies they would like to see in the library (Booth & Guder, 2009). Hayman and Smith (2015) also lay out a structured plan for evidence-based technology inclusion which differentiates “hard” evidence of a need collected through formal assessments and “soft” evidence which is informed by personal
APPLYING SOFTWARE DEV PRODUCT CYCLE

recommendations, anecdotal knowledge from practice, or attention to the work of similar institutions. A formal needs assessment process could take the form of a formal survey or focus group, and an informal process could be based on knowledge of the library and its users, but the evaluation should be holistic and involve representatives from all relevant areas. It should take into consideration the library’s current goals and future priorities, and include assessment of relevant existing systems and users, including both internal and external groups.

The purpose of establishing a need is to have very clear goals and ideas for the technology project. This will make the next phases, research and comparison, more straightforward and constrained. Even if there is no existing technology that meets the outlined need exactly, it is better to begin with a clearly outlined need and work with vendors or a combination of existing technologies to meet that need, rather than trying to fit the need to an existing product. Providing needs and use cases is one of the ways librarians can work to improve library technologies even if they are not engaged in developing their own technologies.

The second phase of the cycle is research. The research phase could be the longest step in this process depending on how many technologies are under consideration. It is important to establish criteria for comparison across different technologies and tailor the research criteria to the individual needs of the library. It is important to exclude features that are not relevant to the problem except in cases where they will provide cost savings or other advantages over technologies already in use. This could be the case, for example, when a new discovery system comes bundled with programs for analyzing usage statistics and could potentially replace those systems already in use.

Examples of criteria that could potentially form the basis of the research phase include accessibility, whether the technology will be able to be used by the population it serves. This criterion could touch on both ADA compliance and issues like the digital divide or mobile divide (Pearce & Rice, 2013). Accessibility might also include the license type and the steps users will
have to take to download or access the technology. Design is closely tied to accessibility; the product should be useable, but aesthetics can also be ranked in this phase.

Most libraries will want to establish cost of the product as a criterion for comparison. The future non-monetary costs, such as training and maintenance, are also worth considering. If a product will need frequent updates or the product will be tricky for patrons to use without significant outreach and training, that should be established in the research phase. It is also useful to get the vendor involved at this stage to answer any questions that might arise during research. The responsiveness of the vendor can be a good indication of how quickly problems with the product will be resolved. Vendor responsiveness can also be considered a criterion for research.

This is also the stage in the process where more traditional collection development and content criteria can be considered. Some considerations might include whether this is the premier product for this discipline or function, whether the content or service is unique and not replicated in the library’s other technologies and platforms, the payment schedule and fees for service, and the audience size for the technology. Both content and format should be thoroughly examined through the lens of the need in this phase in order to facilitate comparison.

The comparison phase of the process will pick up on the criteria established during the research phase. Comparison may also function as a mini cycle with research, because as comparisons are made, more research might need to be done to contrast areas that seem important.

The comparison stage should result in a report or list detailing the pros and cons of each product for easy comparison. If the creator of the report does not hold purchasing power, it is useful to include the comparisons along with a recommendation for purchase. The comparison phase will involve thinking critically about the best option when considering each of the criteria examined in the research phase.
APPLYING SOFTWARE DEV PRODUCT CYCLE

A good example of this is the cost criterion. Depending on the budget of the project, cost alone might make some options more relevant than others, but is worth also considering the non-monetary cost of the product. If a slightly more expensive product will make for an easier transition with less training, it might be more suitable than a less-expensive system that requires a lot of training, data cleaning, or other time costs.

A good example of a library engaged in technology comparison comes from the University of British Columbia (Barsky, Schattman, & Greenwood, 2009). In this article, the criteria are clearly outlined and the two ebook platforms being examined are clearly differentiated in each area. This example also shows that this type of process can be applied to any type of technology, including both content and platform. Another example is this structured comparison of open-source and traditional integrated library systems (Pruett & Choi, 2013). This example has the benefit of comparing several systems with different features and required levels of customization.

Testing will take different forms, depending on the type and audience of your technology. If the audience is patrons, a trial with focus groups or observed testing might help provide valuable feedback that can lead to a decision. If the audience is library employees, testing and training with the intended group might provide the same type of feedback. Library faculty and staff members may have firsthand experience with the technology from their work at other institutions or with outside groups, or may have networks who have this experience, so it may be useful to conduct informal focus groups or reach out to lists and other connections to get testimonials about the usefulness of a product.

When similar products are being compared, testing functionality with students can be very informative. Focus group or observational testing can be used with students to help get real-world perspectives on technologies. There are good examples of testing technology platforms and tools with students and developing frameworks for this kind of testing in the library literature. Bucknell University and Illinois Wesleyan University each compared the EBSCO
APPLYING SOFTWARE DEV PRODUCT CYCLE

Discovery Service to Proquest’s Summon using this technique and contrasted the comparison with observations of students using Google Scholar and a traditional online library catalog. Students were separated into groups and asked to use one of the systems under consideration while librarians observed (Asher, Duke, & Wilson, 2013). Quantitative comparison data was collected, as was qualitative data that could be used to inform setup and student training on these systems. The University of Central Florida also provides good examples of user testing using different methodologies (Kitalong, Hoeppner, & Scharf, 2008). Librarians at Florida International University Libraries analyzed several strategies for usability testing and put a couple to the test in their own website redesign (Dominguez, Hammill, & Brillat, 2015).

Additional resources for planning user testing include Letnikova’s (2008) analysis of 22 publications centered around library website usability testing, in which she provides a framework for writing questions for usability tests. User testing is a very powerful strategy for testing public-facing technologies and can help inform comparison, selection, and maintenance for products. While many of the testing examples focus on development scenarios, testing vendor products, particularly if there are two or more similar products on the market, can help project groups develop an understanding of which products are intuitive for their patrons to use and which technologies they might prefer.

The comparison stage should focus on the need-based criteria and allow the research to lead testing. For example, if the need is a new gate counter that allows remote access to the data, this is the feature that should be tested. It is important not to get bogged down with drawbacks or swayed by perks that do not influence need. While it is important to address products holistically, sometimes features of a particular product, like an unorthodox payment method or a novel access strategy, will sway libraries away from or toward products that are not the best fit for actual needs. It is better to choose the best product and then work with the vendor to correct any difficulties in bringing it to a library audience. Most will be willing to work
with the library project team to develop a strategy and, if not, there is always the opportunity to
go back to the comparison phase with that new information.

The biggest exception to the need-based, tunnel-vision approach is usability studies. If
student testing is used, it will inform the selection of a product, but other data could also be
collected about how students interact with the system in order to inform setup and outreach.
Student-facing systems will frequently also have comparison criteria about ease of student use
and intuitiveness, so collecting data about the way students interact with these products will also
help the team make an informed selection.

Selection will often be the easiest part of the process. It would be a great problem to
have if all the technologies tested are appropriate and of equal quality, but this outcome is
extremely rare. More frequently, the most appropriate product will become clear through the
research, comparison, and testing phases, and selection will be straightforward.

After the relative ease of selection, maintenance might feel like a very long-term project.
Maintenance will be different for different institutions, the product team might be reduced to only
essential personnel or a group might be developed to manage the continuing needs of the
technology. Manuel, Dearnley, and Walton (2010) conducted a survey of library web managers
in the UK that offers several examples of different maintenance structures in institutions. The
most important function of maintenance is to remain alert for new needs that might develop.

The maintenance cycle will go on until a new need arises in the institution, the
technology is no longer meeting the need it was intended to address, or the needs of the
institution have evolved. Whatever the reason, once a technology stops meeting the needs of
the institution, it is time to go through the cycle again. When addressing a new or evolving need,
part of the research and comparison process could be asking the existing vendor to rise to the
occasion, but this also provides the opportunity to investigate newer technologies as well.

While managing the maintenance of a product, it is particularly important to make sure
technology skills don’t bleed out of the institution. The phenomenon of early adopters leaving
the institution without establishing a legacy for maintenance has been observed in educational
technologies (Comber & Lawson, 2013) and is also a frequent problem in libraries. While it is
important to have an advocate for technology projects and a product manager who will help
guide technologies from need to implementation, there also should be administrative
involvement to help ensure that technologies are maintained through the generations of an
organization. Building an emerging or digital technologies role into the organization might be a
good way to do this, but exit counseling and common documentation can also help accomplish
this goal.

Assessing the Product Cycle

Assessing the product cycle can be done after a single product or on an annual or other
regular basis. The criteria for assessment will depend on each library’s overall goals for
implementation. Technology usage or cost per use is a frequently used assessment strategy,
and this can be built into the maintenance part of the cycle to produce regular results. The
assessment strategy can also be bundled with other administrative lookbacks like annual
reports.

Conclusion

Establishing a product cycle can have many benefits for libraries. Helping provide a
framework for technology integration will help inspire innovation toward goals that can benefit all
libraries. It is often easier to stay with vendors and products that have established pedigrees in
the library field, but establishing a formal period of research might help uncover new
technologies, inside and outside of librarianship, that are appropriate for library needs. This
process can also help establish a more communicative relationship with vendors. Being vocal
about the changing needs of libraries will help generate products that are better for libraries.

This process might also inspire some libraries to take matters into their own hands and
establish development teams. Innovations from library-based developers have helped advance
the work of the profession in important ways and modernize library work from the inside out.
APPLYING SOFTWARE DEV PRODUCT CYCLE

The most explicit benefit of this process is that it will help libraries establish the technologies that are not right for them. Just because a technology is new and exciting does not necessarily mean that it will work in any particular organization or help solve a problem. The best way to integrate a technology so it sticks is to begin from a need and work forward to identify technologies that satisfy this need rather than work backwards from the technology to justify inclusion in the library.

Technology integration and any kind of significant change to systems can bring up a lot of emotion in an organization. Sticking to a product cycle can help individuals across the library make clear-headed decisions about the right technologies to invest in and the right time to make technology changes. It is impossible to predict the future of library technologies, but developing a practice based on the unique needs of each library is a good way to ensure that the technologies selected will be successful for your staff and users.
References


