

OCLC on the Responsive Web

Aaron Ganci and John McCullough

Background

Sixty-two. That is the number of widely used mobile or tablet computers on the market as we write this chapter. That number has probably increased in the time it has taken to get to you. The screen on each one of those devices is unique in some way. Some differ in physical dimension. Some have unique aspect ratios. Some have widely different pixel densities. The number of unique devices will become even more extreme in the near future.

Knowing how a digital interface is going to be consumed—at what size, in what setting, with what level of attention, for what purposes, with what input mechanisms—is fundamental to how designers craft good online user experiences. Thanks to the growth and diversification of mobile computing, knowing which contexts your interface will be utilized in is becoming an increasingly difficult task. This has caused a paradigm shift in the way web designers and developers approach their craft. Libraries, of course, inhabit this changing environment.

Like many companies, OCLC has been actively seeking ways to adjust to this new, varied landscape. Our products are frequently the initial web interface for millions of library users around the world. Each of these users accesses our site in a unique way—in either hardware format or usage context. It is important to us that we support as many of the varied devices and situations of our users in the broadest way possible. In redesigning WorldCat Discovery to deliver services into both existing and emerging contexts, we have adopted a responsive web design (RWD) methodology. In this chapter, we outline some challenges we can anticipate in coming years, what advantages we see in RWD instead of other approaches, the impacts RWD has had on our process, and what adjustments we have made to our design and development of our

products. To begin, we'll outline the reasons that led us to adopt a new methodology.

Problem Definition

OCLC's discovery products have not, so far, been widely used on mobile devices, although we designed a mobile app for WorldCat.org available beginning in 2009, and the WorldCat Discovery public beta that launched in 2013 has always been fully responsive to reformat the layout for various screen widths. As of the writing of this chapter, around 9 percent of our page views for WorldCat Discovery come from mobile or tablet devices. This is not a large percentage, but when you extrapolate that out to real usage, it's still around 300,000 page views per week. That's a large—and growing—number of people accessing our sites with smaller devices.

But 9 percent of our total usage might not be enough of our audience to encourage a company as large as OCLC to overhaul its product development methodology. Changing how a large team of people works takes solid rationale and a clear benefit for all parties involved. So what data is out there that convinced our teams to update their approach to design and development and adopt a responsive approach? The answer lies in the emerging research trends and forecasts that point to a dramatic shift in the use of post-PC devices. These estimates reveal an overwhelming trend that people are migrating to tablet and mobile computers faster than ever before. For instance, in the United States, 43 percent of the population—age 16 and up—now own a tablet computer or e-reader.¹ Research in the United Kingdom shows a similar user base, with 34 percent of children ages 5–15 now owning a tablet.² This statistic is especially important to note because it gives a glimpse of the

future. Children who grow up with tablet and mobile computers will carry those usage patterns into their college years and adulthood. It is clear that the population will be moving steadily away from traditional screen sizes. Future users will not be tied to desktop- or laptop-sized monitors to do their work anymore.

The turbulence of this still-emerging transition to mobile devices introduces a number of challenges for design and development teams. Even several years into their use, there are a lot of unanswered questions and changing answers around how these devices fit into our lives. This reality is especially true in the library. Slow adoption by library staff and patrons still leaves many aspects of mobile and tablet usage a mystery. An adaptive approach that opens up our discovery services to mobile devices in the broadest way possible can help us discover how those services get used in mobile contexts over time by analyzing aggregate use.

Before we can look at how the OCLC teams have updated their workflow, it's sobering to take a look at the inherent complexity OCLC faces when displaying library content in dynamic environments.

Library Content and Multitiered Flexibility

OCLC creates highly dynamic interfaces. This is partly due to the nature of our products. We offer customizable interfaces that are used by thousands of different institutions. Each one of our partner institutions has a unique set of characteristics. At the most visible level, each institution may have its own color palette that can be integrated into the interface. At the most basic level, each institution can choose a mix of functionality available to it based on its usage needs (e.g., information about local availability of library materials, interlibrary loan, full-text access, etc.).

The other factor that plays into the flexibility of OCLC's applications is that we're dealing with library content that is widely variable. The content we utilize either comes in a raw format from the MARC (Machine-Readable Cataloging) record or is dependent on the availability of local holdings. When a user runs a search in WorldCat Discovery, we can't predict how big a search result will be, how many authors an item has, how long the title will be, how long a description is, or whether or not an item has cover art. Each item's set of MARC and availability data shapes our user interface differently. To accommodate this, we have always had to think flexibly about how content might be displayed. Now, with the introduction of varied screen sizes, we are forced to add another level of complexity to our responsiveness. The variable nature of our content, coupled with a diverse hardware landscape, presents a very useful case study in flexible design. We will now describe what measures our teams have taken to work in this environment.

Solution Development

We have developed an integrated solution to help improve our work under such variable requirements. There are three important aspects of our approach that we will outline below: content workshops, a UI (user interface) framework, and adaptable usability and utility testing. In sharing these, we hope that you might be able to glean some recommendations about how to work with similar content in a responsive environment. To begin, we will discuss the importance that content plays when working responsively and outline our process in managing content.

Working Content First

A persistent question when working in responsive environments is "How do you manage content?" When we looked at the situation, we found two approaches that we could leverage. One methodology says you should make very few, if any, changes to the content you display to the user. If content is available to the user in one viewport, it should be important enough to show in any viewport. This is a specific flavor of responsive web design called "mobile-first."³ The other approach is more open to showing and hiding content at different views. In this approach, the team must determine—or guess—what content users need at various views. For instance, a user who is accessing WorldCat Discovery on a smartphone may be using it only to quickly recall a small detail. In this case, the user would need to see just a small subset of data about the item rather than a full description. This second approach can help with aspects like page load time because you push only the exact content that is needed. This approach doesn't have a formalized name but generally falls into a methodology called "adaptive web design."

We chose to utilize the first approach because our primary goal for WorldCat Discovery was to create as consistent an experience as possible regardless of viewport. Someone using WorldCat Discovery should be able to use it equally well on a smartphone, tablet, desktop, or any other device. This approach is ideal from a development standpoint because it creates much less complexity. This approach also provides a learning opportunity for our teams. By not fragmenting what we show users, we are able to better track how people are accessing our content on various devices. We will track usage patterns over time and integrate trends from these analytics into future designs.

We were confronted with several problems while putting this approach into action. To use this method, we have to assign priority to content: on smaller screens, we can display less content on one screen, and since we are using near-identical content across all views, we need to make sure that important items are still

visible in small viewports. We discovered that assumptions about what content mattered most varied greatly among the various team members (business vs. development vs. design). Different understandings about these priorities were delaying us because many content disagreements had to be resolved independently.

The solution we developed to avoid these disputes came in the form of something we call “content first workshops.” In these workshops, members from the business, development, user research, and design teams come together before any new designs begin. By meeting early, in the conceptual phase, all of these team members can bring their preferences and concerns to the table and assign content priority together. These sessions are usually led by a UX (user experience) designer who guides the conversation by first asking everyone to agree on user definitions and primary task flows. Making these factors clear in the beginning helps shape the way the team thinks about the content. It is nearly impossible to assign content priority without users and tasks in mind.

With a common understanding, the team is able to look at the potential content that could live on a page, determine what is essential, and prioritize it accordingly. The goal at the end of the session is to have a content-only wireframe of the page in question. In theory, at the end of this workshop, the team has created a semantic view of the page, or in other words what a user might see if CSS is disabled. Even if the content flows into the page in its raw form, it is still accessible and usable. This semantically pure layout becomes an extremely useful tool for the UX and development teams in actually implementing the page. We have found that taking emphasis away from the visual layout through these workshops is an excellent way to work responsively and make sure the design works at any viewport size. In the next section, we’ll discuss a method to visualize the prioritized content.

A Systematized UI Framework

Now that our content is prioritized and universal, we are able to generate one consistent codebase that works regardless of screen sizes. Our next struggle came when we needed to visually style and organize that content. Revising our visual design process, we chose to keep a parallel approach to our content management. However, when presented with the need to generate a design at variable screen dimensions, we discovered important criteria: flexibility and systemization of the visual layer.

When visual designers need to design a page for responsive environments, they tend to think about the interface at various “breakpoints.” What will this page look like on a desktop? Tablet? Smartphone? Recently, this has become more difficult due to the uncertainty of how each of those broad labels is actually defined.

Ideally, visual designers want to customize their layout or elements on the page at each of these breakpoints because they are considering context of use at each screen size (touch formats need larger buttons, bigger text; desktop formats can have wider text columns; etc.). Because the WorldCat Discovery team adopted a mobile-first methodology for our content, it is now improper for the designer to completely overhaul a page layout for specific screen sizes. Remember, we want the utility of the site to be universal. To support this methodology, the visual design must now appear consistent at each screen size but still adapt for usage needs. These restrictions make it cumbersome to specify how elements should be styled at each screen size. Trying to design each individual page and accommodate for all the responsive variables at play actually becomes untenable.

After much struggle, we finally found a solution in the form of a customized UI framework. UI frameworks are essentially visual style guides that systematize a UI design by defining visual and functional details about page elements (buttons, dialog boxes, navigation, etc.). Some very popular frameworks are publically available (Foundation or Bootstrap), but WorldCat Discovery had specific needs that drove us to create a custom framework. We call it CoreUI.

The creation of CoreUI has allowed OCLC’s designers to systematize every aspect of the visual design. This is helpful in a responsive environment because it allows us to simplify how we adjust a UI at various viewports. With CoreUI in place, once we know the size of the screen, we can load predefined elements rather than generating a custom display for every page. For example, in CoreUI, we have several predefined and named button styles (regular, small, expanded). Once we check the screen size through a media query, we simply send the appropriate predefined button style to the page. Instead of guessing what button will work best every time a page is constructed, we define usage details ahead of time. For instance, we know that the “expanded” button (with a width of 100 percent of the screen) works better than a “standard” button at small screen sizes. This workflow allows the designer to systematize what styles will work best at what screen sizes. This process makes developers’ jobs easier because they have clear rules for when to use various iterations of elements. Using the CoreUI has streamlined our process and made it easier to produce a visual design that is consistent yet customized for each screen size. Next, we’ll talk about methods we are using to inform the future of our content strategy and UI implementation.

Usability and Utility Moving Forward

When building any application, especially one as dynamic as WorldCat Discovery, it is important to

test how usable it is for our users. Usability testing has always been a part of our process at OCLC, but it takes new forms in a responsive environment. Usability test sessions usually revolve around a set of tasks that a user will undertake. We recruit representative users to complete these tasks with our design. If the user struggles, we can identify when and why and then adjust our design accordingly. Our process remains similar but has changed slightly now to test at various screen sizes. We now make sure to test appropriate tasks at various screen sizes (usually some combination of mobile, tablet, and desktop) to note alterations that need to happen at each viewport.

For example, in a recent usability study, we investigated how users prefer to facet or filter their results list. We had several questions about when and where we should display the facets at each screen size. Because so much of the mobile and tablet interfaces are new, we often have to make educated guesses about user expectations and then test our solution. At tablet and smaller screen sizes, we decided to hide the facets by default and require the user to click on a button to reveal them. After testing our initial designs, we found that users rely heavily on facets and trust that they are the fastest way to narrow their search accurately. They missed having the facets immediately available; at the tablet view, none of the users were able to initially locate the hidden facets. This told us that we need to adjust our design to make the facets initially visible whenever possible. Only at the smallest viewports do we hide them, and when they are hidden, we make it as obvious and easy as possible for the user to access them. This is just a small example of the importance of usability testing in our new responsive process.

There are still many more unknowns about what tasks our users want to complete with WorldCat Discovery at various screen sizes. To make our product better moving forward, we need to keep refreshing our understanding of appropriate tasks within the different screens sizes. It is safe to conclude that library users' needs will continue to evolve as devices and usage becomes even more varied in the coming years. OCLC is in a unique position to inform this area because we have such a large user and diverse user base across the library sector. Our ability to track usage patterns and compare them to screen size will increase our understanding of how and why people might use a library interface on different devices. The responsive interface that is available today is only a first step in considering what utility WorldCat Discovery will need moving forward. Analytics will either confirm that our universal responsive approach is working or that we need to adapt content more specifically within various devices.

Conclusion

Moving to a responsive web environment is a challenge for any company as large as OCLC. Our challenges are amplified by the inherent complexity of library content and the capabilities of WorldCat Discovery. We have taken several steps to adapt our process to create a better online library experience for librarians and patrons around the world. Our use of content-first workshops, the CoreUI framework, and our ongoing usability and utility testing have helped the OCLC team adapt to the new responsive web. The new methodologies we've put into place help ensure that we can create a consistent experience for our users regardless of what device they are using. However, many questions still remain about what impact these devices have on specific usage patterns of library patrons. Being flexible when it comes to the responsive web has served our team well so far. As we continue to venture into the unknown of the library in the post-PC world, remaining flexible will be more important than ever before.

Notes

1. Lee Rainie and Aaron Smith, "Tablet and E-reader Ownership Update," Pew Research Center, October 18, 2013, www.pewinternet.org/2013/10/18/tablet-and-e-reader-ownership-update.
2. Ofcom, *Children and Parents: Media Use and Attitudes Report*, research document (London: Ofcom, October 2014), 5, http://stakeholders.ofcom.org.uk/binaries/research/media-literacy/media-use-attitudes-14/Childrens_2014_Report.pdf.
3. Luke Wroblewski, *Mobile First* (New York: A Book Apart, 2011).

About the Authors

Aaron Ganci is a user interface designer at OCLC working primarily on WorldCat Discovery. Additionally, he is an assistant professor of Visual Communication Design at the Herron School of Art and Design at Indiana University–Purdue University Indianapolis, where he researches and teaches courses about web design, interaction design, and graphic design. More information can be found at www.aaronganci.com.

John McCullough is the product manager for discovery services at OCLC. As such, he is responsible for WorldCat Discovery Services and has led the product team for its launch. He has worked more than fifteen years in library automation, with a focus on improving end-user discovery. Formerly vice president of Product Strategy at Innovative Interfaces, he oversaw the development of the Encore discovery service. He earned his MLIS at the University of Western Ontario.