Growing Their Own: Legitimate Peripheral Participation for Computational Learning in an Online Fandom Community

Casey Fiesler[1], Shannon Morrison[2], R. Benjamin Shapiro[1], Amy S. Bruckman[3]
[1]University of Colorado Boulder, [2]Syracuse University,
[3]Georgia Institute of Technology
{casey.fiesler, ben.shapiro}@colorado.edu, shmorris@syr.edu, asb@cc.gatech.edu

ABSTRACT
Online communities dedicated to the creation of fanworks (e.g., fiction or art inspired by media such as books or television shows) often serve as communities of practice for learning communication, artistic, and technical skills. In studying one successful fan fiction archive that was designed and built entirely by (predominantly women) fans, we observed processes of legitimate peripheral participation (LPP) in which some of these fans began in peripheral roles and came to be more involved in the technical aspects of the archive over time. In addition to outlining positive outcomes, we discuss the challenges of supporting learning within this CoP, particularly with respect to the burden on experts. We discuss potential implications and solutions for the problem of expert scarcity in CoPs, and propose that LPP within fan communities can be leveraged for broadening participation in computing among women.

Author Keywords
broadening participation in computing; communities of practice; computing education; fandom; fanfiction; learning; legitimate peripheral participation; online communities; open source

ACM Classification Keywords
H.5.m. Information interfaces and presentation (e.g., HCI): Miscellaneous

INTRODUCTION

How are we going to get enough people to not just build this thing, but maintain it… We're going to have to grow our own. – Naomi (participant interview)

CSCW '17, February 25-March 01, 2017, Portland, OR, USA

ACM 978-1-4503-4335-0/17/03…$15.00
DOI: http://dx.doi.org/10.1145/2998181.2998210

Could a community of people, finding that existing technology is not meeting their needs, unite to build their own system from scratch? This would be a resource-intensive process in both time and expertise. The endeavor would be even more surprising if the group of user-developers were members of a group typically underrepresented in the computing field, and more surprising still if many of these developers began as completely novice programmers. Yet this is what happened in the creation of Archive of Our Own (AO3), a successful fan fiction archive. This community of fan creators really did "grow their own"—not only their own software, but also their own developer base, and thus had success both in designing an online community for fan creators and in creating a community for peer-facilitated computational learning. In this paper, we discuss both the successes and challenges of this unusual case study of peer learning.

Fandom communities come together over a shared love of media such as books or television shows, and have long been recognized as spaces for informal learning [6,22]. In the act of creating *fanworks* (fiction, video, art, and webpages inspired by and dedicated to the things they love, from *Harry Potter* to *Star Wars* and everything in between) these fans often learn complex technical and communication skills. This learning takes place inside affinity spaces [5] or communities of practice [43], where community members share a passion for fan creation and improve their skills through interaction with one another.

Communities of practice (CoP) exhibit three inter-related characteristics: (1) participants are *mutually engaged*, i.e. they work together within relationships that entail assistance of one another (e.g., mentoring); (2) they share a *joint enterprise*, a sense of common purpose or product that they work together toward; and (3) they co-develop a *shared repertoire* of resources, including common domain knowledge (e.g. of the Harry Potter universe, of storytelling techniques, or of HTML) [43]. One process by which newcomers learn to become productive members of these communities of practice is called legitimate peripheral participation (LPP) [28]. Recognized in prior work as a

valuable process for learning within many different kinds of online communities [1,9,27,37], LPP can enable learning within close-knit fandom CoPs as an important part of developing literacy and writing skills [5,6,10].

In studying the design and use of AO3 as an example of one such community, we observed LPP in action. We also saw how the AO3 CoP enabled some of its members to learn computational skills, a proficiency not traditionally associated with fan fiction. AO3, which boasts over 1 million users and 2 million fan-created works, was designed and developed by a group of mostly women to support the values and needs of their community [16,21]. Within an environment that prioritized training community members to do technical work over bringing in outsiders, fan volunteers had opportunities to learn new computational and design skills and to contribute to AO3, in part through processes of LPP within the structure of a CoP.

In order to examine the mechanisms at work within AO3 from both a user and developer perspective, we conducted interviews with 28 members of the AO3 community. With an emphasis on what we learned from 11 participants directly involved with the design, development, and running of the archive, we discuss both the successes and challenges of this particular learning environment. In addition to showcasing the positive learning outcomes of LPP processes within AO3 development, our data also highlight an important challenge associated with LPP: the potential for overburdened experts and training bottlenecks when key knowledge within the CoP is concentrated within a few individuals.

The contributions of this work include: (1) a case study of LPP in action (where some fans began by working on less technical tasks) that could inform designing for learning within existing close-knit communities; (2) a conceptualization of a particular challenge associated with this learning environment, that of overburdened experts; and (3) proposals for potential solutions to this challenge that could apply to other CoPs. Finally, we also present AO3 as a successful example of organically sparking interest in computational learning among an underrepresented group, with suggestions for how fan communities might be further leveraged to broaden participation in computing among women.

BACKGROUND

Fan fiction is a prominent form of fanwork, consisting of written stories based on existing media properties. The history of fan fiction dates back to Sherlock Holmes and Jane Austen, but it found a large community starting with *Star Trek* decades ago and subsequently flourished online [21]. An excerpt from author Naomi Novik's testimony before Congress (on the topic of fair use) perfectly describes fanworks as well as the community that surrounds them [45]:

> *[B]efore I wrote one word of my first novel, I wrote fan fiction, built online computer games, wrote open source archiving software, and created remix videos. I met hundreds of other artists creating their own work and found an enthusiastic audience who gave feedback and advice and help… We were gathering around a campfire. We were singing, telling stories with our friends. The campfire was just a bigger campfire, thanks to the Internet, and instead of telling new stories about Robin Hood, we told new stories about Captain Picard, because that is who we saw on our television every week.*

Novik, an award-winning novelist, was also the lead developer in the early days of AO3, having written a blog post that sparked the idea for the site. It launched in 2008 after poor experiences with other online fan fiction platforms led to a rallying cry for an online space of their own [16,19,21]. Though AO3's code is open source (and anyone can submit patches, where a new feature or piece of code could be incorporated into the whole), nearly all of the design, development, and maintenance comes from fan volunteers who also participate in its story authoring community. Matching the demographics of fan creation communities generally [22], the majority of these volunteers have been women, who are (at least in recent decades) underrepresented in computing fields generally [4], and even more so within open source projects [36,42].

Beyond contributing to the design and writing the code, there are many other tasks that rely on volunteers and are essential to maintaining the archive. In addition to the core Accessibility, Design, and Technology (ADT) committee and a subcommittee of testers, a small army of "tag wranglers" serve an important function for AO3 by manually curating and connecting the complex user-created folksonomy that underlies the archive's search structure [16,24]. Other volunteers serve on committees that handle, for example, abuse and support, legal matters, and external and internal communication, both for AO3 and for the broader non-profit that supports it, the Organization for Transformative Works (OTW). Though the core development team consists of approximately a dozen people at any given point, there are hundreds of volunteers that actively help run AO3.

Related Works

In a community of practice (CoP), a group is focused on a shared goal, task, or practice [43]—in the case of AO3, this practice is the design, building, and support of the archive and its surrounding community. Legitimate peripheral participation (LPP) is one process by which *newcomers* can become part of a CoP, a social, apprenticeship-based mechanism for supporting learning [28]. The "peripheral" part of LPP refers to how newcomers may initially participate in non-critical tasks (requiring minimal knowledge but allowing observation of expert practices and community structure) and then work their way toward more integral community involvement as they learn. It is a way

of "learning by doing" and also learning by observing, in which the learning is situated within the social context of the practice. As a result, newcomers learn in part by directly observing experts' practices. As they do so, they become mutually engaged in the joint enterprise and develop mastery of the shared repertoire of the CoP. This movement from periphery to center, and from novice to expert, is called *centripetal participation*.

Online communities have long been considered potential spaces for CoP and LPP [23]. A number of researchers in this space have examined how newcomers become engaged in a community through processes of LPP [1,9,27,37] or contexts in which LPP might improve community building [32,35]. Wikipedia is one commonly cited example of LPP, with prior work showing how new wiki editors become more involved over time [9] (often moving from a content-centric perspective to a community-centric perspective) and even how reading can be a form of legitimate participation [1]. However, it is also important to consider how the success of LPP within these communities is shaped by other characteristics of CoP, including the distribution of critical domain knowledge within the community (i.e. how deeply it is part of the *shared* repertoire) and the extent to which community members actually engage in joint enterprises. These factors may shape how observable experts' practices are, and consequently, how successfully LPP processes permit centripetal participation (i.e., learning) by community members.

With respect to learning programming skills, researchers have used LPP as a framework for describing learning processes within open source communities [12,14,26]. In fact, the construction of positive social relations through LPP processes is predictive of sustained participation in an open source community; the authors of one study posited that these social relations are as important a creation as the software product [14]. This work suggests that some forms of mutual engagement (at least those based in a discourse of encouragement and mentorship) are more effective than others. However, prior work has also shown that open source projects sometimes have negative aspects to their cultures and social relations, particularly with respect to gender dynamics (at least partially to blame for the very low proportion of women in these communities) [36].

Value differences are cited as one reason for lower participation of women in computing [4]. Even beyond occasional open misogyny in open source communities [36], values of meritocracy and authority enforced by acrimony can keep women away [29]. Prior work has also shown that in open source communities, non-technical contributions are not as valued as technical ones [11]. Nafus notes that despite values of openness in open source that should attract women, this is often instantiated in ways that present troubling authorship norms and delegitimize certain types of social ties [36]. Instead, norms of politeness and low barriers to entry are more likely to retain women in open source projects [34].

Computational learning is also an increasingly social activity, particularly in constructionist approaches to learning that include creation and sharing of artifacts [25]. The creators of the Scratch community cite the influence of fandom scholar Henry Jenkins [22] as inspiration for thinking about remixing as a pathway to computational learning through LPP [17,33]. Jenkins, among others, has presented fan creation activities as a powerful model of informal learning as well, particularly for literacy skills [6,22]. Within the space of HCI and online communities, researchers have characterized fandoms as surprisingly close-knit communities with strong social norms [15,20], i.e., sites of mutual engagement.

Researchers have also explored how fandom's status as a networked affinity space affords it the opportunity for members to enjoy unique and diverse community-based mentored learning [5,10]. Describing "distributed mentorship" among fan fiction writers, Campbell et al. introduced an informal, distributed, weakly tied form of mentorship that mirrors the feedback mechanisms traditional in fandom communities. They found that allowing the difficulties of mentorship to be shouldered by a wide array of temporary mentors with varying levels of expertise allows authors to receive valuable feedback without overburdening any single mentor [10].

In her work describing English language learning through fan fiction communities, Rebecca Black describes fandom as an affinity space based on shared passion and interest, rather than a community of practice [5]. She notes that in affinity spaces, the role of "expert" and "novice" are variable and context-dependent. Halverson also argues against characterizing fandom as a community of practice, because learning trajectories are not hierarchical [18]. However, unlike fandom communities more generally, the development and maintenance of AO3's software infrastructure is more rooted in a specific task or practice with more clearly delineated levels of expertise (similar to other open source projects). The AO3 software development sub-community could therefore be seen as an open source CoP *inside* a fan fiction CoP *inside* broader affinity spaces (fan fiction culture overall, various fandoms, and even various web development communities). The context of the surrounding culture and existing values of fandom affinity spaces therefore has the opportunity to shape the learning culture taking place within the CoP.

In the current study, we use AO3 as a case study to unpack evidence of computational learning within AO3 as a CoP, show how LPP supported learning within this CoP, and how its LPP practices may present challenges to ongoing sustainability and scalability.

METHODS

In studying the design and use of AO3, we conducted semi-structured, in-depth interviews [40] with 28 participants involved with the archive: 6 developers/designers, 5 staffers (2 on the communications team, 2 tag wranglers, 1 member of the OTW legal committee), and 17 users. We recruited these participants through (1) direct contact with the current AO3 development team (asking for participant volunteers as well as pointers to past developers, whom we also reached out to directly), and (2) a public recruitment post on Tumblr targeted at AO3 users. In recruiting we did not screen for any particular attitudes or types of experiences with respect to AO3. Our recruitment efforts were successful in part because of a level of trust within the community; the first author has been involved with the umbrella OTW organization (though not directly with AO3) for a number of years, and the second author identifies as a member of fandom communities.

We conducted interviews via phone/Skype (18), in person (2), or through online instant message (7) for those who did not feel comfortable with voice communication. Though we spoke to all participants about learning in fandom generally, our data about learning processes within AO3 come largely from the 11 interviews with those directly involved with the archive (6 developers/designers and 5 staffers). All of these 11 participants identified as women, which though unsurprising given typical gender representation within fan communities [22], is in stark contrast to the open source world generally, which is typically less than 10% women [36,42]. Demographics were not part of our recruitment efforts, though our participants told us that the core development team has always been nearly entirely women. All of our participants were adults (ranging 23-62), and the majority self-identified as white. Though most of our participants live in the United States, four of the six developers we interviewed are outside the U.S. (Canada, Australia, and Europe). Table 1 breaks down the roles of the 11 participants who are directly involved with AO3 or OTW. The remaining 17 participants are all current active *users* of AO3 who do not have a role in its functioning.

Name	Position	Status
A*	Development	Active
Betsy	Development	Active
Lucy*	Development	Former
Maia*	Development	Former
Michele*	Development	Former
Naomi*	Development/Founder	Former
Heidi*	Legal Committee	Active
Kimberly	Tag Wrangler	Active
Maureen	Communications	Active
Mira*	Communications	Active
mmmdraco*	Tag Wrangler	Active

Table 1: AO3-involved participants, with their role and current status (whether or not they were still actively involved in that role at the time they were interviewed)

With respect to this table and the labeled participant quotes in our analysis section, it is sometimes appropriate to use participants' real names (or chosen pseudonyms) in research publications, particularly when they are creators who deserve credit for their work [8]. We gave our participants this option. Names marked in Table 1 with a * are real names or pseudonyms they chose themselves; all others are anonymizations we created.

In addition to general questions about the design, development, and functioning of the archive and its teams of volunteers, we also asked participants when they started using AO3 and/or how they first came to be involved with the organization, what kinds of projects they worked on, and how learning took place. We also asked all participants about their experiences learning new skills through participation in fandom.

Two independent coders first conducted an inductive, open-ended thematic analysis, identifying emerging themes from our interview data [7], then came together for discussion and synthesis. In addition to interviews, we also looked to related documentation, including public discussions about the archive's development from blog posts as well as the public code repository, bug reports, and feature requests, in order to supplement knowledge about processes at work within AO3.

Through our analysis process, an additional major theme emerged—the design of the archive as feminist HCI [16]—but here we focus on another major theme, that of learning and of AO3 as a community of practice.

LPP PROCESSES IN AO3

Throughout our interviews with those who worked on the design and development of AO3, there was evidence of processes indicative of LPP, including movement from simple tasks to more complex ones (typically associated with a transition from newcomer to oldtimer), learning through observation, and help from the community. We also saw evidence of positive learning outcomes related to these experiences.

Newcomers to Oldtimers

Though there were core developers in the early days of the archive (described by one of our participants as "three women writing code in an apartment together"), most who have contributed to AO3's code came into the project later. The most common story we heard from developers we interviewed (both their own, and relating the experiences of others) was of fans volunteering to help with the archive due to their passion for fan fiction or AO3's mission.

I thought the archive was a great idea, I wanted it to become a reality, and I thought I would be able to help. I wanted to learn Ruby on Rails because I was interested in it but probably wouldn't have been able to learn on my own. OTW [the non-profit that supports AO3] gave me the possibility to learn the technical skills and in turn I would help build the archive. – Betsy (current dev team)

This passion for the fan community itself was also essential to the functioning of the archive, because the early developers prioritized training fans to help as opposed to bringing in outside developers (for example, from the broader open source community). If these fan volunteers came in with programming experience, they might immediately start working on development. Others began in an even more peripheral position in AO3 and then slowly become more involved as they observed how things worked and picked up technical skills.

For example, many AO3 volunteers who come in as tag wranglers do so because they are highly invested in a specific fandom (e.g., they spend their time "wrangling" *Harry Potter* stories because they love *Harry Potter*). They may then volunteer to wrangle more obscure fandoms simply because someone needs to do it and they are invested more generally in helping to maintain AO3. Then in some cases they will move from tag wrangling into other archive tasks—even working on development. This is similar to how Wikipedia editors often start on content-centric tasks due to a specific interest but then move to more community-centric tasks [9]. One participant described tag wrangling and support duties on AO3 as "gateways" into more mission critical work on the archive.

It's not difficult once you are a tag wrangler to really be involved with the whole process. If you do get to that position, you go, "Well, actually, I'm really good at coding." It makes it easier to get in there and move to another position or help out in multiple things. – Kimberly (tag wrangler)

Also, particularly in the early days of development, potential developers would be actively recruited from their more peripheral activities in the community—for example, archive users who provided feedback or bug reports.

We recruited a lot of fannish development. I was pretty relentless with that feedback form in the early days. I'd say, "Hey, I love your work. I really appreciate that you took the time to do this, but would you like to get more involved?" We'd actually pick up a couple of developers because they were silly enough to leave good feedback. – Maia (former dev team)

Testing can catalyze increased interest in developing computing skills [13], and that appears to have occurred organically within AO3 as well. A number of developers started out by testing, and then moved to actually writing code, including one of our interview participants:

I expressed interest in joining as a testing volunteer. I was inducted as a tester, had a poke around, looked at the code out of interest, then was taught how to actually submit a fix—as in, typos or similarly easy stuff. Then I was hanging out a lot in chat with the AO3 coders and testers… but eventually I was asked to join the committee staff. – A (current dev team)

In situations like these in which new coders needed to be trained, they would be given easy tasks, and then build up to more difficult and critical tasks. Lucy, a former member of the development team, differentiated between tasks for "baby coders" (simple enough for a novice, and also more peripheral and self contained) and tasks for those with more experience who would work on code more directly related to critical archive functions. The first type of task would, with time and growing expertise, lead into the second.

There were a lot of us at different levels. We started with a very basic project, building a blog…. And the bookmarks feature was originally built by a baby coder, because it seemed like a vaguely doable thing for someone with not much experience. Then we started to do stuff like commit code… building the Archive. – Lucy (current dev team)

Finally, these once-newcomers would eventually become oldtimers in the community, finding themselves training new coders the way they themselves had entered.

It was really empowering because I'd come in with zero experience, and there was just an attitude that all of us could learn and could be helped. Naomi did a lot of the training early on. Later, I did some training myself. – Lucy

Learning in a Community of Practice

The learning that took place during this process of becoming more involved with the technical work on the archive was largely learning-by-doing, supplemented by formal training. In addition to current volunteers running small chat room "classes" online to teach programming skills, new coders would be grouped with experienced ones to learn "on the job." A few of our interview participants described this as "sharing" knowledge.

If you wanted to be a volunteer to help them, they'll teach you how to code the programs they're using. I thought that was really marvelous and I love all the sharing that goes on. – Mira (communications staff)

When they did the first call for volunteers they specified that you don't need to know programming. They had a whole thing where volunteers were put into groups based on their knowledge level and the senior coders helped out the junior coders. – Betsy

This ethos of sharing and helping was dominant in how learning took place. There was a persistent chat room for the development team, and someone was almost always there to aid newcomers (though this itself presents additional challenges, discussed later).

We'd just hang out in chat and try things out, and if we got stuck someone with more experience would help. And when we started building the Archive, they would give us something to do and we'd just bash away at it until we figured it out, with help. - Lucy

Learning from others, asking for help, and working in collaboration was the common refrain of participants even

when discussing other functions that make the archive run. For example, tag wranglers also discussed working with each other to make decisions, and asking for help when needed.

There is a very close community just for the tag wranglers, because AO3, for all of their different volunteer stuff, there are these special kinds of chat rooms that they have there on Campfire. Once you get in there and start talking, you can just jump in there and go, "Okay, I have a tag that's popped up and I don't really know what to do with it. So, here. Help me figure it out." - Kimberly

Participants expressed that this culture of helping and sharing as part of learning came out of the larger fandom community, and so was part of the AO3 development culture from the start. When we spoke to AO3 users about their experiences learning in fandom outside the context of the archive, we heard similar stories, particularly with respect to picking up non-programming technical skills like HTML or video editing. It is common practice to ask other fans for help, to learn by seeing others' work, or to work on projects together as a way of improving their own skills. One participant explained her feelings about the type of learning that takes place in fandom communities versus in more formal learning contexts, emphasizing how the community aspect contributes to a positive learning environment:

Writing stories, or editing videos, or creating layouts, you get a different sort of feedback than you do in a professional community. People are allowing you to fail and to make mistakes. They know that it's a labor of love. So, they'll encourage you to get better without being like, "You're doing terrible. This is what you need to do. This is the right way of doing things." It's a little more loving. – Kirsten (user)

The atmosphere described here is a contrast to some descriptions of open source communities where decision-making is often acrimonious [36]. However, with respect to AO3, the two types of authorship on the site (fan fiction and code) are related in an existing underlying value system of supportive mentorship. Therefore, novice learning may be easier within AO3 development than in other open source projects—because the CoP borrows from trust established from the surrounding community, and therefore participants feel comfortable reaching out for help in a way that they might not in a more acrimonious environment.

Impacts of Learning Beyond AO3

Every AO3 developer interviewed had either a story of their own, or about someone else, wherein the skills that they learned helped them outside of the archive or even on the job market. For example, working on AO3 gave Maia project management experience that she leveraged in her career. Another coder who started out working on the archive as a class internship learned enough programming skills to get a job after university. Even when programming was not directly related to their careers, volunteers suggested that their work on the archive had a positive impact:

Having worked kind of on the back end of [AO3] some, I really understand how to utilize a lot of the features that I think even some of the more seasoned just regular users don't totally understand. Having that knowledge on the back end kind of helps me. – Kimberly

It was clear from our interviews that the community aspect of learning was essential. For learners, the "we're all in this together!" atmosphere of sharing and helping made them feel like important, contributing members of a project and community that they cared about. Even archive users who have not contributed directly to its development or maintenance expressed a feeling of belonging and ownership in the project—another example of Antin's finding on Wikipedia that *readers* are not free-riders, but instead important participants in a community [1].

A common theme from our participants when talking about learning new skills (technical or otherwise) was that it arose out of necessity, so that they could do the things they loved. One mentioned a huge amount of historical research she'd done to write a particular fan fiction. Another told us about becoming a Final Cut Pro wiz because she wanted to make fanvids. Another credited her participation in fandom as impetus for her to learn Python even though the learning itself did not take place within the community:

I taught myself a lot of Python because I wanted to program a stupid little program that would like list different characters and give me like a random pairing. So I pretty much taught myself to program just for that. Because fandom was there, it gave me incentive.– Tara (user)

In fact, more than half of our interview participants mentioned that they learned or improved their HTML or other web design skills because of their participation in fandom (often to format fan fiction for posting online). One participant, who ran a *Harry Potter* fan fiction archive in the early 2000s, credited work on that site for the success of many volunteers in their later careers.

We've had some amazing people come through Fiction Alley on the writer's side, the beta reader's side, the coder's side. But if I listed where they are now, there are people who are at Tumblr. There are people at movie production companies. There's a writer over at Buzzfeed. There are people at Facebook. At Google we have seven people who are former Fiction Alley coders. That's awesome! – Heidi (legal committee)

Overall, within both AO3 and fandom communities generally, the culture of both building up *people* and building up the archive was described as empowering:

The ethos was very much on empowering fans. Partly by bringing together people who had those skills already, and there were lots of those people. And partly by passing on

those skills. So we were building the community's skills as well as building the archive itself. – Lucy

CHALLENGES OF LPP

Despite the positive outcomes, these same processes of community-based learning are not without their challenges. In particular, though it seems to have been universally positive for *learners*, this CoP can place a burden on experienced coders. One problem was that the most experienced coders were perpetually in short supply. It was a priority from the beginning of AO3 that the site be built *for* fans *by* fans, truly an archive of their *own* [16].

"How are we going to get enough people to not just build this thing, but maintain it?" So, from the very beginning, we thought, "All right. We're going to have to sort of teach people internally. We're going to have to grow our own." – Naomi (founder, former dev team)

Though this culture of "growing our own" was positive in many ways, including building up a sense of community and safety that can be especially important in fan spaces [21], it also meant that the most experienced coders could be overburdened with the combination of coding workload and helping to guide novices, sometimes leading to burnout. Two of the former developers we spoke to cited this reason for why they left the team, despite their positive feelings about the archive and community. The same "chat room presence" that learners cited as so helpful also could feel overly demanding to experienced members of the team.

It was constantly like, "Can you hang out in the coder campfire chat room?" I had a full-time job, which at the time was going through a particularly stressful phase… A big part of why I walked away was that I was being asked to prioritize the needs of the novice coders over my own time. – Michele (former dev team)

A lot of [teaching new coders] was about providing structure. But it was also about being in the chat room and being in the tester chat room and saying, "Right. Thank you everyone for being here at this bizarre hour. Don't worry. I'm not sleeping either." – Maia

Therefore, though LPP processes were successful from the learners' point of view, the lack of formal infrastructure around training presented challenges, despite the best intentions from the community oldtimers.

We also had developers who also had a deep well of kindness, and were available to be kind to people, and teach them, and help them where they could, and keep that rolling. But I don't think we ever focused enough on support mechanisms and creating an environment where there was an instructive way to learn enough to really sustain that. – Maia

The current developers we interviewed recognized this problem as well. The difficulty in training up new coders resulted in a persistent lack of experts. As a result, much of the effort of the development team is spent on maintenance ("fixing things that break," as one participant put it) rather than new development.

We're not actively working on [direct messaging, a commonly requested feature] right now, because we need to rewrite the inbox code before we can do that. It's on our roadmap and we want to do it, but I think it will be a while before that happens. Right now we pretty much only have the resources to keep the Archive up and running. [We need] people with the time and the coding knowledge to implement new features. - Betsy

Keeping AO3 up and running also requires more and more resources as the archive (and its code) has become more complex over time—a complexity that is compounded by increased expectations of an enthusiastic and ever-expanding user base.

We built the basic thing surprisingly fast. But once you have people actually using it, there's a high technical burden in keeping it working… and it's partly that it's big and established now. When something is small and new everyone expects glitches. – Lucy

Also as a result of increasing complexity, there are less and less of the small, easy tasks that were originally given to "baby coders." Thus there have been fewer *new* coders in recent years than there were in the beginning when a large part of this learning and training was taking place.

It got harder as we went along. I remember the point where the Archive code was so small that I had a fair understanding of what each bit did. But as it grew and got more complex, that was less true, so it was harder for new people. – Lucy

Naomi, the lead developer in archive's early days, recognizes the trade-offs inherent in this decision to keep AO3 development "in-house." She acknowledged the current challenges in maintaining a development team when burnout has become increasingly common:

I do think in retrospect, that's probably where the blame actually falls. It started to be much harder to train new people and bring new people in after the archive actually went open to the public. Before then when it was just us in the chat room making a site for ourselves, it was much easier and much more fun to help newbies and to walk them through with it afterwards… I'm very glad that we tried. But at the same time, it's definitely not like there's a constant pipeline for new people coming in and getting trained. - Naomi

Though currently anyone can submit a pull request for AO3, which is technically open source, the inner circle of major developers remains all fan volunteers. One former developer we spoke to worries that the site as it grows may become unsustainable. She suggests that it may become necessary to seek out coders, particularly women, who may not already be heavily involved in fandom.

It would be nice if things got fixed… there are people who are coders who may be interested in working on something like this. I think that particularly given all the stuff happening in the last couple of years in my field around women in technology, that there is a large open source project with more than 85% female contributors, it is baffling that it is hard to contribute to. – Michele

Another member of the current development team told us that an ideal solution would be to hire a fan who is also a professional programmer, in order to free up the main team for working on new features. In other words, they see this as a potential solution to the *technical debt* [41] that has built up over time due to increasing code complexity.

Our current hope is to get someone under contract, and throw money at her and go, fix this, fix that, tell those people how stuff works, yay! We also considered contracting out coding projects and refactoring code and cleaning up code to free up our brains for training and more design thought and stuff. – A

Despite the growing challenge of maintaining an expert developer base large enough to deal with the complexities of the archive, AO3 has been successful and is growing faster than ever. It launched in 2008, and in 2014 hit the mark of 1 million works; at the end of 2015, it hit 2 million.[1] Many of the users we interviewed commented that the archive must take a lot of work to run, but did not have any conception of the challenges they face.

DISCUSSION

Other researchers have examined open source projects as CoPs with learning taking place through LPP [12,14,26]. AO3's software development CoP presents an example of LPP around learning programming skills that is different from these in several ways: (1) it is predominantly women, in contrast to most open source projects; (2) much of the learning taking place is with completely novice coders (as opposed to programmers with some skills learning more); and (3) it is situated within a larger community of interest where other types of learning are also taking place.

This last point, that this is a community where the software development exists within a larger community, is also a contrast to open source, where the community comes together *around* that particular project. Though some software projects might develop their own culture, AO3 began with the culture/community first and in fact designed the system around the community's existing values and needs (as well as creating a sub-community to perform the building that instantiates those same values) [16]. This community orientation presents both the greatest advantages and the greatest challenges to LPP. On the one hand, the ethos of sharing, helping, and learning-from-others comes in part from the existing ties that learners have to this community in which they are heavily invested. It also maintains the sense of the archive being *their own*, something that was extremely important to the genesis of the project. On the other hand, the decision to keep development *within* the existing community has contributed to a bottleneck in training and burden on experts.

Though researchers have used LPP as a framework to understand the success of online communities, it is also important to consider the ways in which communities enacting LPP can struggle or fail, and how those struggles relate to relationships between LPP and other organizational structures and processes. In considering how what we learned from AO3 might apply to other learning communities, we present a conceptualization of the overburdened expert challenge, as well as a set of design implications that could better support this kind of LPP. However, also taking into account the successful components of LPP within AO3, we end with lessons and implications for broadening participation in computing.

Conceptualizing the Bottleneck

Developer scarcity as a problem is not unique to AO3, and open source projects often consider the "truck factor" as how many developers would have to be hit by a truck in order for the project to be incapacitated [2]. However, the bottleneck within AO3 is one specifically related to learning, in that there are not enough experts to support novice learners that exist at a level not common in open source, and the learning processes within AO3 were unable to close this expertise gap. Therefore, not only is the potentially low "truck factor" a problem, but also the potential for continued burnout of experts caused in part by mentorship burden.

In contrast to the work environments studied by Lave and Wenger [28], the voluntary nature of AO3 (and open source generally) adds this new challenge—peer production communities tend to have a higher rate of dropouts (hence, the truck factor). In Lave and Wenger's case studies, novices have opportunities to observe work, and experts facilitate their understanding of it—but the experts' primary role remains enacting the core domain work of organization. However, in our data, experts reported that mentoring could *distract* from doing that core work. This is similar to Luther's studies of online creative collaboration, detailing the burden often placed on the *leaders* of volunteer creative projects [30]. He suggests distributed project leadership as a solution to overburdening, where more team members take on responsibilities. Similarly, LPP ideally supports a *gradient* centripetal pathway from less expert to more, so that responsibilities are not all concentrated at the top. This progressive development is what sustains the CoP even as experts depart.

In peer production projects, individuals self-select for the tasks for which they will be the best contributor [3]. In LPP, much of what determines which tasks one might be best suited for is that level of expertise. However, as learning

[1] https://archiveofourown.org/admin_posts/366; https://archiveofourown.org/admin_posts/4385

takes place it is important to consider not just someone's *current* level but also what will help them reach the next level of expertise. Vygotskian theory suggests that one should select tasks within their *zone of proximal development*, which represents the distance between actual ability and potential ability in interaction with more capable peers [44].

In theory, the knowledge gradient is a resource for reducing the burden on experts, because novices' peripheral work does not always require the supervision of experts, only near peers who can provide feedback and guidance. These near peers can have ample capacity to extend each others' zones of proximal development, even while they are not yet experts. They do have enough expertise to contribute that could advance the novice's development; therefore, full experts aren't necessary to support learning.

Therefore, in the case of AO3 a core cause of the bottleneck could be insufficient interaction with near peers. Our participants told us about the tutoring of beginners, but not as much about beginners learning from slightly-less-beginner peers. Beginners may also have had insufficient opportunity to *observe* more advanced practice, from near peers or experts. This would limit the extent to which the technical repertoire of expert AO3 coders becomes a shared repertoire within the CoP, and increase the extent to which time-consuming additional teaching activities were necessary.

Solutions for Easing the Burden on Experts

Having identified some challenges that AO3 faces within their CoP, one implication to this work is to consider solutions. How could communities like AO3 support learning in a way that is not overly burdensome to experts, and what might these challenges also suggest for supporting learning in other online communities? Based on what we learned from both the successes and challenges of AO3, we propose four design implications that could also apply to LPP more broadly: (1) providing mechanisms for indirectly observing contextual expert practice; (2) focusing on intermediate retention and supporting gradient skills matching; (3) supporting community identity construction along with skills learning; and (4) leveraging communities with similar value structures to create inter-communities of practice.

The scarcity problem with AO3 developers is in contrast to the distributed mentorship network of fan fiction writers described by Campbell et al. [10], and that was present in the overall AO3 writing community. Some of the aspects of distributed mentorship that they describe are based on the public and persistent nature of feedback and mentorship. These concepts, very different from the chat room atmosphere described by some of our participants, could provide a starting place for how to help create more structured training. Simply having expert work take place in a persistent and novice-observable manner could support learning while imposing minimal burden on experts. Creating mechanisms for novice observation of expert practice might simultaneously improve novices' learning and reduce the burden on experts. For example, in large open source projects, mailing lists can serve this purpose, as experts discussions about the merits of different approaches to problem solving (such as about which of several proposed patches to accept).

Supporting the *gradient* skills pathway is also important, and one method might be to pay special attention to avoiding intermediate attrition—that is, finding ways to retain those with intermediate skills who will be the *near peers* to novices. It will be important to consider whether there are appropriate development tasks available for each of these gradient levels, and for matching slightly more advanced AO3 programmers with novices who could learn from them. Other open source projects will sometimes mark open tasks with levels of difficulty, but in AO3 task assignment as described by our participants is done in a more ad hoc way. Building scaffolding for task selection into the system, perhaps similar to Luther's suggestions for scaffolding task distribution in collaboration [30], could help support a gradient skill distribution.

In addition to better infrastructure to support retention, another solution might be to recruit more experts (or even more at intermediate levels who could serve as near-peers). One participant suggested recruiting outside developers, despite their lack of ties to the community. It is worth considering, then, whether there could be LPP processes around learning to be *part of the community*, rather than learning to code. Learning structures within fandom generally do not tend to be hierarchical [18], and unlike accounts of some open source communities where non-technical contributions are de-valued [11], AO3 emphasizes the value of non-development contributors such as tag wranglers. Therefore, though we see clear LPP processes within the technical stream of AO3 development, other types of learning occurring could also benefit the CoP as a whole. Just as Black notes that "expert" versus "novice" can be context-dependent in a fandom affinity space [5], perhaps novice programmers could serve as expert fans to help teach norms and values of the community to outsider, expert programmers. Within AO3, the identity construction component of LPP that is often highly important (such as in open source communities [14]) is largely moot since even newcomers already identify strongly with the broader fandom community. But given how closely tied the technical architecture of AO3 is to the values of the community [16], working on archive development in collaboration with fans has the potential to result in this type of identity construction.

In easing the expertise bottleneck, beyond (1) retaining and training more experts from within the community, and (2) bringing in more experts from outside the community, a final option might be (3) connecting to experts in neighboring communities. AO3 is unusual though not

entirely unique among software projects, and so there could be opportunity to create an even broader CoP around developers in online communities that share similar values. For example, the blogging community Dreamwidth is a volunteer-run open source project with a large number of women developers and a significant fandom user base [32]. Could these sites with shared values form networks for peer production that are orthogonal to their core purposes but that leverage their commonalities of mutual engagement? Members of AO3, Dreamwidth, and similar communities may actually have much in common in terms of norms for constructive interaction, even though their shared repertoires diverge. These overlaps could create contexts for alliance across sites, with mechanisms for joint training, including building features and even sharing code. Thus these *inter-communities of practice* could add to an ethos of sharing and help shoulder each other's training burdens.

Implications for Broadening Participation

Another implication of this work is to consider how the successful aspects of LPP within AO3 could be leveraged in other contexts. Broadening participation in computing among women is an area of interest within the computing research community, and AO3 provides a unique example not only of a system designed and built mostly by women but also of one way that women have *learned* computing skills. Prior work has shown that context can be important for engaging women in computer science, particularly with respect to finding personally meaningful projects [31]. Fandom communities have long been a pathway for learning technical skills—traditionally things like HTML, web design, and video editing. Our work shows that this same interest in fandom could potentially be leveraged to spark interest in and teach programming as well.

In analyzing AO3 both in terms of online community design and as an environment for learning, the importance of "growing their own" cannot be overstated. Even beyond the context of fan fiction, something else that makes AO3 unique is that it is a system designed *by* and *for* the community that it serves. Though increasing community identification is traditionally part of the LPP process, in this case the inherent strong sense of community identity (and shared values) contributed both to the success of the site and to the positive learning environment. Therefore, another pathway into successful community-based computational learning could be to encourage communities to create their own technologies. For example, the website Ravelry, an online community for knitters, was designed by a knitter to meet the needs of that community (though was built by a non-knitter developer) [38].

Within open source communities, building something that you need is often called "scratching an itch" [39]. The difference here is that when software developers decide they need a new piece of software, they already have the skills required to build it. This process becomes more challenging when the group who needs something are under-represented in computing and therefore may lack an existing expert development base.

AO3 did have an advantage in that there were a few very dedicated, very talented programmers who spearheaded the process and therefore served as experts to usher in novices with mentoring and training. However, with the right infrastructure (as emphasized above), "seeding" community development projects with invested experts could also create the right kind of learning environment. Moreover, they could also potentially be supported by the *inter-communities of practice* that we described.

CONCLUSION

We have presented AO3 both as an interesting case study of legitimate peripheral participation within a software development community, and as an organic example of sparking an interest in computational learning among women. In contrast to other open source projects, AO3's software development CoP exists within a larger community (fandom) that came with a highly ingrained value system. This community orientation presented both the greatest advantages and the greatest challenges to LPP. Existing ties to the community led to an ethos of sharing, helping, and learning-from-others that worked well for learners, and also maintained the sense of the archive being *their own*, which learners described as empowering. However, the decision to keep development *within* the existing community also contributed to a bottleneck in training and a burden on experts.

The difficulty that AO3 has faced in maintaining an expert coder base suggests an important challenge associated with LPP—the potential for a self-perpetuating cycle of over-burdened experts. This potential pitfall should be a consideration in designing for community-based learning, where more purposeful infrastructure for training could potentially be informed by concepts of distributed mentorship [10] that have been successful elsewhere in fandom communities. We also suggest that lessons learned from AO3's success could be deliberately leveraged to provide a meaningful context for developing technical skill for members of groups traditionally under-represented in the computing field, as well as serve as a model for community groups seeking to develop software to better meet their own needs.

ACKNOWLEDGMENTS

Our thanks to reviewers and colleagues for their helpful feedback on this work. We also thank the fan fiction and AO3/OTW community, not only our research participants but also those who volunteered or shared our call for participation. This work was funded by NSF IIS Award #1216347.

REFERENCES

1. Judd Antin and Coye Cheshire. 2010. Readers are not free-riders. In *Proceedings of the ACM Conference on Computer Supported Cooperative*

*Work & Social Computing (*CSCW).

2. Guilherme Avelino, Leonardo Passos, Andre Hora, and Marco Tulio Valente. 2016. A Novel Approach for Estimating Truck Factors. *24th International Conference on Program Comprehension* (ICPC)
3. Yochai Benkler. 2002. Coase's Penguin, or, Linux and the Nature of the Firm. *Yale Law Journal* 112, 3, 367–445.
4. Sylvia Beyer. 2014. Why are women underrepresented in Computer Science? Gender differences in stereotypes, self- efficacy, values, and interests and predictors of future CS course-taking and grades. *Computer Science Education* 24, 2–3, 153–192.
5. Rebecca W. Black. 2007. Digital Design: English Langauge Learners and Reader Reviews in Online Fiction. In *A New Literacies Sampler*, Michele Knobel and Colin Lankshear (eds.). Peter Lang Publishing, New York, NY, 115–136.
6. Rebecca W. Black. 2009. English-Language Learners, Fan Communities, and 21st-Century Skills. *Journal of Adolescent & Adult Literacy* 52, 8, 688–697.
7. Virginia Braun and Victoria Clarke. 2006. Using thematic analysis in psychology. *Qualitative Research in Psychology* 3, 2, 77–101.
8. Amy Bruckman, Kurt Luther, and Casey Fiesler. 2015. When Should We Use Real Names in Published Accounts of Internet Research? In *Digital Research Confidential*, Ezster Hargittai and Christian Sandvig (eds.). MIT Press, Cambridge, MA.
9. Susan L. Bryant, Andrea Forte, and AmyS. Bruckman. 2005. Becoming Wikipedian: transformation of participation in a collaborative online encyclopedia. *Proceedings of the ACM Conference on Supporting Group Work* (GROUP).
10. Julie Ann Campbell, Sarah Evans, Cecilia Aragon, Abigail Evans, Katie Davis, and David P Randall. 2016. Thousands of Positive Reviews: Distributed Mentoring in Online Fan Communities. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work & Social Computing* (CSCW).
11. E. Gabriella Coleman. 2013. *Coding Freedom: The Ethics and Aesthetics of Hacking*. Princeton University Press, Princeton, NJ.
12. Donald Davis and Iffat Jabeen. 2011. Learning in the GNU/Linux community. *Proceedings of the Conference on Information Technology Education* (SIGITE).
13. Betsy James Disalvo, Mark Guzdail, Tom Mcklin, et al. 2009. Glitch Game Testers : African American Men Breaking Open the Console. *Proceedings of Digital Games Research Association* (DiGRA).
14. Yulin Fang and Derrick Neufeld. 2009. Understanding Sustained Participation in Open Source Software Projects. *Journal of Management Information Systems* 25, 4, 9–50.
15. Casey Fiesler and Amy S. Bruckman. 2014. Remixers' understandings of fair use online. *Proceedings of the ACM Conference on Computer Supported Cooperative Work & Social Computing* (CSCW).
16. Casey Fiesler, Shannon Morrison, and Amy S. Bruckman. 2016. An Archive of Their Own: A Case Study of Feminist HCI and Values in Design. *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI).
17. Sayamindu Dasgupta, William Hale, Andrés Monroy-Hernández, and Benjamin Mako Hill. 2016. Remixing as a Pathway to Computational Thinking. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work & Social Computing* (CSCW).
18. Erica Halverson. 2007. Reality television, fan behavior, and online communities of practice. *Science Education*, 244–246.
19. Karen Hellekson. 2015. Making Use Of: The Gift, Commerce, and Fans. *Cinema Journal* 4, 3, 125–131.
20. Serena Hillman, Jason Procyk, and Carman Neustaedter. 2014. " alksjdf ; lksfd ": Tumblr and the Fandom User Experience. *Proceedings of the Conference on Designing Interactive Systems* (DIS).
21. Anne Jamison. 2013. *Fic: Why Fanfiction is Taking Over the World*. Smart Pop, Dallas, TX.
22. Henry Jenkins. 2006. *Convergence Culture*. NYU Press. New York, NY.
23. Christopher M Johnson. 2001. A survey of current research on online communities of practice. *The Internet and Higher Education* 4, 1, 45–60.
24. Shannon Fay Johnson. 2014. Fan fiction metadata creation and utilization within fan fiction archives: Three primary models. *Transformative Works and Cultures* 17.
25. Yasmin B. Kafai and Quinn Burke. 2013. The social turn in K-12 programming. *Proceeding of the 44th ACM Technical Symposium on Computer Science Education* (SIGCSE).

26. Kouichi Kishida and Yunwen Ye. 2003. Toward an understanding of the motivation of open source software developers. *Proceedings of the International Conference on Software Engineering* (ICSE).

27. Cliff Lampe and Erik Johnston. 2005. Follow the (slash) dot: Effects of feedback on new members in an online community. In *Proceedings of the ACM Conference on Supporting Group Work* (GROUP).

28. Jean Lave and Etienne Wenger. 1991. *Situated learning: Legitimate peripheral participation*. Cambridge University Press, Cambridge.

29. James Leach, Dawn Nafus, and Bernhard Krieger. 2009. Freedom Imagined: Morality and Aesthetics in Open Source Software. *Ethnos* 74, 1, 51–71.

30. Kurt Luther, Casey Fiesler, and Amy S. Bruckman. 2013. Redistributing Leadership in Online Creative Collaboration. In *Proceedings of the ACM Conference on Computer-Supported Cooperative Work & Social Computing* (CSCW).

31. Jane Margolis, Jean J. Ryoo, Cueponcaxochitl D.M. Sandoval, Joanna Goode, and Gail Chapman. 2012. Beyond Access: Broadening Participation in High School Computer Science. *ACM Inroads* 3, 4, 72–78.

32. Christina Masden and W. Keith Edwards. 2015. Understanding the Role of Community in Online Dating. *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI).

33. Andrés Monroy-hernández and Mitchel Resnick. 2008. Empowering Kids to Create and Share Programmable Media. *Interactions*, 50–53.

34. Eunyoung Moon. 2013. Gendered Patterns of Politeness in Free / Libre Open Source Software Development. *Proceedings of the 46th Hawaii International Conference on System Sciences* (HICSS).

35. Gabriel Mugar, Katie Devries, Kevin Crowston, et al. 2014. Planet Hunters and Seafloor Explorers: Legitimate Peripheral Participation Through Practice Proxies in Online Citizen Science. *Proceedings of the ACM Conference on Computer-Supported Cooperative Work & Social Computing* (CSCW).

36. Dawn Nafus. 2012. "Patches don"t have gender': What is not open in open source software. *New Media & Society* 14, 4, 669–683.

37. Anthony Papargyris and Angeliki Poulymenakou. 2005. Learning to fly in persistent digital worlds: The case of Massively Multiplayer Online Role Playing Games. *ACM SIGGROUP Bulletin* 25, 1, 41–49.

38. Linda Polin, Sheila Pisa, Bonnie Nardi, Sylvie Noel, and Dev Sen. 2015. Mediated Search for Information: Finding Information and Help Inside and Outside Ravelry.com. *The 2015 Annual Meeting of the American Educational Research Association*.

39. Eric S. Raymond. 2001. The cathedral and the bazaar: musings on Linux and open source by an accidental revolutionary. O'Reilly Media.

40. Irving Seidman. 1998. *Interviewing as Qualitative Research*. Teachers College Press.

41. Edith Tom, Aybuke Aurum, and Richard Vidgen. 2013. An Exploration of Technical Debt. *Journal of Systems and Software* 18, 6, 1498–1516.

42. Bogdan Vasilescu, Daryl Posnett, Baishakhi Ray, et al. 2015. Gender and Tenure Diversity in GitHub Teams. *Proceedings of the ACM Conference on Human Factors in Computing Systems* (CHI).

43. Etienne Wenger. 1998. *Communities of Practice: Learning, Meaning, and Identity*. Cambridge University Press, Cambridge.

44. J.V. Wertsch. 1988. *Vygotsky and the Social Formation of Mind*. Harvard University Press, Cambridge, MA.

45. 2014. The Scope of Fair Use. *Hearing Before the Subcommitte on Courts, Intellectual Property, and the Internet* (January 28 2014). DOI=113-82-86454-.pdf