Configuring Discourse: A Critical Reading of How-To Guides for Designing and Managing Online Communities

Communication and interaction on the web is mediated through layers of software and protocols. As we increasingly turn to study communication, interaction and learning on the web, it is essential that we look below the surface textual interactions and consider the interfaces, functionality, designs and the ideas of those who administrate and build those systems. Through analysis of excerpts from how-to guides for running and administering online community software, this essay offers a point of entry into the discourse of design and configuration that plays an integral role in the software layer of online interaction. This is an argument for why and an example of how we can use this kind of how-to literature to help study software in a way that respects the tension that exists between the material affordances of software with the dynamic and social nature of software as a component in symbolic interaction. Configuring discourse is evocative of two elements of the essay. First, it represents the way that online community software configures online discourse, setting the terms for interaction in design and functionality. Second, it suggests the discourse of configuration that exists in the how-to books, where administrators and developers lay out the tactics for how to configure this software in accordance with their goals and values.

Researchers are increasingly exploring the textual records of online community sites, web forums and more broadly sites where individuals can discuss, publish share and interact with each other, as contexts where young and old alike are developing valuable skills and sharing and building knowledge [1]. Researchers have documented hundreds of instances of individuals using web forums to learn a variety of professional and leisure skills: composition, visual art, providing and receiving criticism, and a range of other skills. As more social and political action occurs in online communities it becomes critical to understand not only what people are saying to each other online, but also how the systems they interact with are shaping what they say.

In establishing a research method for netnography, ethnographic research of online communities, ethnographer Robert Kozinats suggests that the wide availability of archival data in old discussion boards and listservs provides the digital equivalent of "access to recordings of every public social contact in a given culture".[2] While he recognizes that the "the nature of the interaction is altered—both constrained and liberated—by the specific nature and rules of the technological medium in which it is carried" [3] this idea that the textual records of content from online discussion boards and other mediated communications are the same as transcripts of public conversations is problematic. Kozinats method of research seems to have stepped away from Miller & Statler's earlier work on online ethnography which suggested that researchers understand the Internet as a shifting assemblage of technologies. In their words, the goal of ethnographic work on online communities required the need "to 'disaggregate' the Internet; not to look at a monolithic medium called 'the Internet," but rather at a range of practices, software and hardware technologies, modes of representation and interaction that may or may not be interrelated by participants, machines or programs [4]. Exploring how various software components come together to shape and structure that participation is essential for making sense of this textual record.

Studying the textual record of interactions on web forums and related online community sites is valuable. However, that value is limited by a lack of understanding of the server side-software that makes such records possible. There are some excellent exceptions to this point, like Yochai Benkler's extensive and nuanced discussion of the architecture and design of Slashdot in *The Wealth of Networks* and work analyzing discursive interactions between Wikipedia editors [5]. However,we lack an understanding of how particular kinds of discussion board software and other custom designed community software features structure and configure online discourse. We also lack an understanding of the goals and values that developers and administrators bring into the design, configuration and administration of online community software. While there are numerous ways to approach studying this subject, for example interviewing software developers and site administrators or conducting ethnographic research, there is also considerable insight to be gleaned from studying the guidebooks created by and for online community site developers and administrators. This study is first and foremost an exploration of what it is that we can learn and theorize from the guidance in these types of popular and widely available how-to books. To that end, the goal of this essay is not to provide a definitive argument about online community software but to provide potential avenues for thinking about online community as evident in these texts and suggest ways to think about the value of how-to books as source material for software studies more generally.

In what follows, I first explain the value of studying how-to guides for software studies. From there, I analyze selections from three how-to guides. Powazek's *Design for Community: the art of connecting real people in virtual places* released in 2001; O'Keefe's 2008 book, *Managing Online Forums: Everything You Need to Know to Create and Run Successful Community Discussion Boards;* and Buss and Strauss's 2009 book, *Online Communities Handbook: Building Your Business and Brand on the Web.* Through this analysis I provide considerations for those interested in studying online communities who wish to more deeply consider how developers and administrators have shaped that discourse. I also discuss the implications of this approach for understanding the nature of online community software.

Defining the Online Community Software Layer

The user experience of participating in an online community –posting comments, engaging in a threaded discussion, building out your profile page– is made possible by server side software. As Monfort and Bogost suggest, the study of software involves the study of layers of software on top of software intertwined with particular pieces of hardware [6]. The software layer implicated in these how-to books is the software that users are directly interacting with on the server when they participate in online community websites. Often this software comes in the form of applications like PhPbb, Vanilla Forums, the Ultimate Bulletin Board System, or vBullitan. In other situations it is made up of custom PhP, Python, or Perl scripts that interact with server side databases. While we experience the web as a series of individual pages that load one at a time in our web browsers, and many of us understand that web pages can be individually written in HTML, most of our interactions with the web are actually mediated by these other software packages and the tweaks, hacks and plugins that site administrators use to change that software. These above named software systems establish rules for what users can and can't do, what users can and can't see, and how others in an online community site see users and their online actions. In short, the indirect object of analysis is the layer of software that exists just below the surface (from the user's point of view) and on the servers that we interact with.

The Value of Studying Discourse in How-To Guides as Software Studies

Software, like all technology, does not exist in a vacuum. Software is created, deployed, and managed by individuals and organizations toward their own set of goals. The subject of analysis in this particular study is popular how-to guides on running and managing this software. How-to guides as offer a point of entry for understanding the theories of users, of design, and the values which are prevalent in an ongoing discourse about what this software should do.

These texts can be understood and analyzed as according to Fairclough and Gee's approaches to the study of discourse [7]. While the study of discourse is generally associated with Foucault's work on power and institutions, Gee and Fairclough are more broadly concerned with connecting that approach with sociolinguistic approaches to understanding how we do things with words. In this tradition, part of discourse involves what Gee refers to as "cultural models," the storylines that makes sense of our individual situated meanings. In this respect, guide books describe particular functionality of software and explain when that functionality is and is not appropriate toward a given set of goals. This approach to discourse is not focused on revealing social or cultural forces, but on how individuals use texts for "world building" – that is how people use words and texts to do things. In Gee's words, "People use language to communicate, cooperate, help others, and build things like marriages, reputations, and institutions. They also use it to lie, advantage themselves, harm people, and destroy things like marriages, reputations, and institutions" [8]. In this sense, the objective is to something akin to Goffman's facework, in which we develop ways of thinking about how individuals and software systems participate in the performitive nature of online social interaction and identity [9]. How people do things with words online and how those online community systems interact in those actions. The stories and the explanations of particular tactics and techniques in these books suggest particular goals, values and ways of thinking about users and I am interested in thinking about what these words tell us and what these words do.

In this respect, how-to guides are valuable to the study of software in several ways. As published works these guidebooks represent thoughtful and reflective organizations of knowledge and experience. These books present the well-developed working theories of those with technical experience administrating, developing, and designing this software. They do not contain the typical users' ideas or understandings of these systems. Instead, these books are accounts of individuals who have likely spent much more time thinking through and developing a theory of their experience. Beyond this, such texts are actual places in which the cultural script for a given technology is being defined and distributed. That is, given that the meaning and values surrounding a given technology always come with a certain amount of ambiguity, these how-to guides represent activities to make sense of and manage that ambiguity. The fact that there is a market for these kinds of guides is indicative of the fact that there is an audience for them, one that is, at least to an extent, receptive to the ideas and approaches in them. With this said, it is not necessarily that these texts shape or define larger cultural scripts. As the reflections and ideas of working practitioners the content of these guide books is similarly shaped by the working ideas within the community of software designers and administrators. To this end, this essay is not about identifying where these ideas come from, but instead about using these books as points of entry into discussion of how particular functionality in community software is deployed and what values are evident in how it is deployed.

By focusing on how these texts frame, present, and suggest the value of particular elements of the technical components of software, we can explore the interplay between discourse and technology. I will suggest that these configuration stories—stories of configuring software systems and stories of configuring online communication and discourse itself—suggest a range of valuable lessons for understanding the nature of online discourse. Throughout this analysis, I work to suggest the value for researchers in using an awareness of how online community is itself configured to better understand online communities.

Three Configuration Stories

What follows is a close reading of selections from three how-to guides. Derek Powazek's *Design for Community: the art of connecting real people in virtual places* released in 2001 Patrick O'Keefe's 2008 book, *Managing Online Forums: Everything You Need to Know to Create and Run Successful Community Discussion Boards* and Anna Buss and Nancy Strauss's 2009 book, *Online Communities Handbook: Building Your Business and Brand on the Web.* These three books were selected from a larger set of 15 books which serve as the basis for a larger study. These three were selected because they each offer particularly compelling and diverse points of entry into developer and administrator discourse.

In this essay, the notion of “web community” is itself defined by the way that books about web communities describe them. For Powazek, this includes any kinds of social interactions over the world wide web, although the book is primarily focused on asynchronous interactions like commenting. For O’Keefe, this is about online forums, primarily those running software applications like phpBB or Vbulletin. For Buss and Strauss online communities are defined as “Web 2.0 sites such as wikis, message boards and blogs”[10]. Together, these definitions focus on asynchronous text based modes of discursive interaction on the web.

The goal here is not to be comprehensive, but rather to engage with sections of each of these books in order to illustrate the value of this approach, isolate some important factors for consideration in the study of online communication, and to begin to work, from the bottom up, toward an understanding of social interaction and conversation in online communities’ relationship to the technical and functional layer of the software. In Powazak’s, case I will present his overarching description of what online community software does and engage in depth with a tactic he suggests called burying the post button. In O’Keefe’s case, I will present a series of tactics he offers for what he calls “banning users and dealing with chaos” as this material provides substantive ways for thinking about the extent and the limits of technical control. Finally, I will walk through how Buss and Strauss explain online community users as commodities. These commodified users, described from a behaviorist psychological perspective, establish the basis for valuing virtual rewards and reputation systems as tools for creating and sustaining social hierarchies.

The goal of this essay is not to be comprehensive; instead, the goal is to provide an initial tour of what an ongoing and critical engagement with this literature can tell us about software that frequently includes the words “social” and “community” in its description.

Bury the Post Button: Page Layout, Empowerment and Manipulation

Derek Powazek’s *Design for Community: the art of connecting real people in virtual places* published in 2002, is one of the first major books which focuses on how to add what he refers to as “community features” to websites. Published by New Riders, a publisher of popular technical books on web design, the book includes a range of prescriptive advice. Drawing on Powazak’s experience developing the sites fray.com and kvetch.com as well has his experience working as a consultant for Netscape, Lotus, and Sony the book provides a technical audience of designers with tactics and theory for the use and value of a range of what he describes as “community features”. Along with the prescriptive advice for how to implement and design ways for end users to interact with websites, the book includes ten interviews with an impressive array of individuals working on online community software development (?) in 2001. The book represents an interesting moment in history. While much has happened and changed in the last 10 years the book remains a foundational text which presents a particular perspective on the design of web forums, and discussion and commenting components enabled in a range of different sites. The book is widely accessible, it is on the shelves of at least 210 of the world’s libraries who participate in WorldCat. While the book offers a range of engaging points to consider, my intention here is to focus in on how Powazek frames the mission of designing for community and then to explore how that mission is and is not evident in the details of how he suggests that designers actually implement that vision in particular designs.

In the following quote, from the book's introduction we find one of the central tensions that exist in the idea of enabling what Powazak describes as "community features:"

> This is all about power. Giving your users tools to communicate is giving them the power. But we're not talking about all the tools they could possibly want. We're talking about carefully crafted experiences, conservatively proportioned for maximum impact.[11]

While studies of discourse often turn to explanations based on power and control it is not generally explained and expressed so forcefully by those we study. There is an important tension between the first two sentences of this quote and the last two. In the first sentences, Powazek is focused on empowering users. In these sentences the "tools to communicate" are all about empowering users, about handing over control. This is part of a central idea in many of these texts, that old media was a controlled experience in which producers produced and consumers consumed. With that said, the last sentences are also all about power, but not at all about empowering users. The end of the quote is all about the soft power of the designer. The act of control in this case comes through deciding what tools one is going to give their theoretical users, how one will allow them to communicate. Importantly, this is explicitly about not about"all the tools they could possibly want," that is, empowering users is not an attempt to give them everything they want. In the final sentence we find out exactly what it means to be a designer of community in this text. In this case, the designer creates "carefully crafted experiences" the experience of being a part of the online community is itself a designed thing. That experience is itself designed for "maximum impact" where the idea of maximization itself implies a vision of enabling specific kinds of communication between particular kinds of users. Throughout all the texts on online community design management and configuration the tension between empowering and giving up control to users and controlling, manipulating or exploiting them is ever-present.

Throughout the book there are a range of places to explore how these ideas fit with Powazak's prescriptions for those interested in building sites and platforms that enable community interactions. Digging into one of these ideas in depth will help to explicate how this view of the relationship between users and designers is embodied and enacted in design. The following excerpts come from his section Rule 2#: Bury the Post Button.

> In my experience with community features, I have observed a proportional relationship to the distance that the post button is from the front door of the site and the quality of the conversation on the site. The farther away it is, the better it gets. [12]

The post button is intentionally placed to provoke a specific kind of conversation from a particular kind of user. At the end of the first sentence we find what he sees as the value that "community features provide." In his case, the value he is trying to optimize is the "quality of the conversation" something that he will further explain to us shortly. With that goal in mind, he offers us a theory of visual design that will strike many as counter intuitive. To explain this idea he draws us in to think about the metaphorical "front door" of the site. Like many of these texts and works on web design in general, the layout of a building is used as a way of describing the experience of moving through a site. In doing so, he, and many of these texts, spatialize sites.

What was a series of files, or a series of linked documents becomes a home or a building. One can read into this the implication that we aren't having intimate, or quality conversation on the front porch or in the parlor.

With this said, if the goal of your site is to engage in dialog why would one want to "bury the post button"? The point goes against much of the common wisdom of web design exemplified in work like Krug's "Don't Make Me Think: A Common Sense Approach to Web Usability." Why is Powazack suggesting to obfuscate and make it more difficult for a user to do what they want to do? He goes on to explain:

> Why would this be? Because, in this case, the multiple clicks it takes to read the whole store are actually a great screening mechanism. Users who are looking for trouble or aren't really engaged in your content will be put off by the distance. They lose interest and drift away. But the users who are engaged by the content and interested in the results of the conversation will stick with it. These are the people you want to retain, because they're much more likely to post great thoughts. [13]

Making it more difficult for a user to get to the post button and respond and share their ideas is part of an explicit attempt to generate a particular kind of discussion. Here we find out a bit more about what it is that Powazak thinks counts as quality conversation. It has to do with separating out different kinds of users. There are "users looking for trouble" and users who "aren't really engaged" as kinds of people he wants to filter out. In that process he hopes to retain the "users who are engaged by the content and interested in the results of the conversation." In short, Powazak has in his mind a set of categories of good and bad users and the design decisions he is providing are intended to result in a particular kind of discussion between the good kind of users.

In this case, the design and relationship of pages, making someone click though multiple pages is the instrument, or the tactic, he presents to configure discourse. He goes on, however, to explain this "can translate into different things when applied practically." As an example he suggests "perhaps the best place for the call to action ("Post your thoughts!") is at the bottom of the page instead of the top." Notice here that the distance metaphor, how far away from the front door the individual was, has now shifted from clicks to scrolling down the page. He explains "That way at least your users will have had to skim through some content before they are given the chance to respond." So distance operates as both a function of clicks and of page layout. Both the design of the structure of a site with "community features" and the visual design of the individual pages is being explicitly explained in terms of structuring both of these features to create particular kinds of discourse and dialog.

While this may seem like a somewhat self-evident point, of course designers are designing according to their goals. However, this presents serious implications for what anyone who studies conversations and discussions in online communities can say about what the textual records on a page of a particular online discussion can tell us. Any interpretation of online discourse needs to start with the recognition that, in all likelihood, the site has been designed to invite and engage particular kinds of people in particular kinds of discussion.

Here the specifics of this description become important. One suggestion for researchers is to directly consider the point Powazak proposes, that this is a particular design tactic and that if one wants to engage in a study of online discourse it is probably a good idea to look at where the post button is as the placement and location of the call to action is likely to act as a filter. Understanding exactly how that filter works and who is being filtered out and in is always going to be a tricky game. However, it is essential to realize, at the base level, that the designers of sites are using visual design and information architecture in an attempt to prompt particular kinds of people to particular kinds of actions and discussions.

Dealing with Problem Users, Bans and Chaos

Patrick O'Keefe's 2008 book, *Managing Online Forums: Everything You Need to Know to Create and Run Successful Community Discussion Boards* offers readers a wide range of advice and suggestions for doing exactly what the title of the book suggests. The book is the result ofO'Keefe's years of experience running, managing and administrating an array of online forums, including everything from SportsForums.net, KarateForums.com, phpBBHacks.com, CommunityAdmins.com, and PhotoshopForums.com. Where Powazek's book is more broadly about designing for a range of potential emerging community features O'Keefe's is very directly focused on running, administrating, and managing a particular species of online community site, the web forum. In his case, he primarily discusses the functionality of two of the most widely used web PHP and MySQL-based platforms, vBulitin and phpBB.

While there is a considerable amount of content in the book that is potentially of interest, one section of the book provides additional discussion of problem users and the kinds of tactics and strategies that one might invoke to curb and control their behaviors. The chapter titled "Banning Users and Dealing with Chaos" describes a range of problem users and how to deal with them. The problem users include everything from "adverquetions" in which new users show up and offer thinly veiled marketing messages, to "content thieves" who repost forum content elsewhere, to users like the "reply-to-every-post-guy", the "freedom of speech guy" who insists that the "freedom of speech entitles people to say whatever they wish, whenever they wish, wherever they wish" which he explains remains "one of the most common misconceptions and problem issues for community administrators." Other types of users described include the "I'm Creating My Own YourSite.com" user and the most intense "Hate Him, My Minions! Hate Him!" in which someone who runs a competing online community site "becomes jealous of you and abuses his position to manipulate his user base" and sends them all attack your site [14]. O'Keefe has provided us with an inventory of a range of problem users. He goes on to suggest exactly how to structure and manage forums to deal with these problem users.

Curbing Abuse: Report buttons and Automating Facework

After providing a range of suggestions for how to handle particularly difficult user situations, O'Keefe starts by suggesting that a post reporting system is a great way to curb abuse. In this case, having your forums include a "report post" button next to each post that will add the post to a queue for the administrator or other site managers to review. Beyond including the button, he encourages admins to encourage members to use the report button and make sure that moderators use their judgment in deciding when to remove posts for violating the guidelines for the discussion board. Creating and posting these kinds of guidelines gets its own chapter in both Powazak and O'Keefe's books. There is a technical system and an emergent social system in place here. First there are the structural components, the report post button, and the queue of reported posts; the second requires getting participants in the discussion boards to use the report post button and finding and recruiting moderators who will read the queue and use their judgment in deciding what is and is not a violation. This mixture of a technical system and social norms, in effect, implements a particular set of ideas about governance. Anyone can report anyone, and moderators judge. Already significant to the study of online discourse is the fact that moderators delete and prune discussion on the site. Discussion threads are not direct transcripts of conversation -- they can and often do change over time, particularly if they are on hot-button subjects.

O'Keefe then explains how "helpful notices" can affect posting. For example, on his phpBBHacks.com support forum site, whenever a user starts a new post they are prompted with a notice in red text right above the subject line where they title their post which urges them to make sure that they are posting in the right section of the forums. As another example of a helpful notice, he suggests the value of prompting users who respond to discussion threads that are older than a specified age that, again in red text. "The thread is X months old and that he might want to consider creating a new thread instead" [15]. Where the report post button in Powazek's book attempts to police content in the discussion and potentially remove inappropriate posts, the helpful notices suggested by O'Keefe act to pre-police posts. The goal here is to influence the poster at the moment they are about to post by giving them a particular bit of just-in-time guidance. In the case of the later suggestion, this guidance can be programmed to only appear in particular kinds of discussions. These cues are being consciously deployed by administrators to shape discussion. It's essential that anyone who wants to study communication in these kinds of sites think about how things like the post box, or the posting page itself, may include this kind of just-in-time information in an effort to steer conversation in a particular direction.

Under the heading of "innovative tools" he goes on to explain two examples of ways that a developer friend helped him by creating hacks for web forums he manages. These hacks are interesting on a few levels, principally the way they illuminate what he refers to as "automation" can directly affect the nature of online discourse. The hacks also offer a way to understand how guidelines, norms and rules of a community site can be enacted as procedural or algorithmic rules. These scripts and hacks (O'Keefe uses the term Hack to refer to extensions or plugins for software like phpBB) effectively become actors in the communicative discourse of the forum. They lay in wait and pop out at prescribed moments in discursive interaction to mediate and perturb the order of the communicative act. In this sense, the web forum, and it's hacks are something akin to robots participating in and altering the kinds of discursive interactions Goffmen would describe as facework. For example, consider how O'Keefe describes a particular problem on his phpBBHacks.com site. He explains,

> We had used the word censor to block out inappropriate language but I was thinking about that system one day and it dawned on me: What if those posts were stopped when the user tried to post them? And what if the community software explained why and even highlighted the sections of the posts where the violation(s) occurred, allowing users to make adjustments without losing their post. [16]

The word censor he describes is functionality that will block out words from an inappropriate word list when displayed on the site. Instead of obscuring censored terms, O'Keefe wanted his site to automatically reject posts with censored words. He then wanted to provide in context information about what terms had triggered the censor. As a result, all communication on the site first involves a brief inquisition from the site's censor. A post will either pass or fail, and if it fails the user can make changes to what they want to say before it is ever recorded. Where the previous censoring tool wouldn't display words on the censor list, this new plugin won't let even traces of them remain. O'Keefe goes on to provide us with the text prompt he gives users who trigger the word censor.

> Your post has triggered our word censor feature. The portions of the post that triggered the censor are highlighted in the preview below. Please adjust it and attempt to post again. Please note that abbreviating the term/string that was censored or circumventing our word censor feature in any way constitutes a violation of our user guidelines, and your post will be removed. [17]

When you tell the user what is being censored they can easily work around it. Instead of writing "ass" you write "a$$" and you have tricked the word censor. This note is included with the reference community guidelines, to wrap a normative layer around the word censor functionality. Earlier in the book, when describing the concept of basic built in word censoring functionality, he suggested the importance of this norming layer: "Don't forget, you can never censor every vulgar term. Don't even try. It's not possible." He explains that "People will use words or come up with new ones that you didn't or can't censor and you will have to remove their posts." The lesson is, "do not institute an 'if it's not censored, it's OK' type of guideline" [18]. O'Keefe wants to use the word censor to help automate part of the rules and norms of the site, but for him, it is critical that the automated functionality not become the rules and norms of the site. His hack both exposes the logic of the censor to the user, inviting them to revise their comments but at the same time explaining to them exactly how they could circumvent the rules. For O'Keefe, this is where the social contract of the site's guidelines becomes critical. The guidelines ensure that users don't game the censor and if they do those users invite harsher reaction.

This brings up some significant considerations for studying discourse in these online communities. For O'Keefe the benefit of this approach is clear, "This saves us time and it saves the member time—his post doesn't have to be removed and we don't have to document the violation and contact him, because the violation is never made." [19] There is a benefit to both him and the community member, they both save time and avoid an altercation. With that said, there is no record of what happened here. The user attempted to say one thing, the system politely asked them not to say it in a particular way, the user self-censored based on that feedback, and all that remains it the result of this interaction. In short, when this kind of functionality is enabled on a site we are studying something that isn't so much a transcript of what was intended to be said but instead a transcript of a conversation which was pre-censored at the point of origination.

Difficulties of banning and new levels of manipulation

From here, O'Keefe describes several approaches for banning troublesome users. As he explains

"Idiots and bad people exist and you'll be dealing with them." For him, the process of banning a user is something that the user bring upon themselves. "As an administrator, you are simply reacting to what a member does." [20] Most of what he says about banning is what one might expect; he identifies particularly egregious individual situations which might result in needing to ban a user and discusses minor ways that a user might repeatedly violate the community guidelines in such a way that they ultimately should be banned.

O'Keefe methods for banning and reflections on those methods, are critical for understanding how control functions in these web forums. Spesifically, =how admins shape online discourse from the technical level. O'Keefe first describes banning usernames. Banning a username keeps a particular user from posting to the forum. Here the site administrator wields considerable control and power. They can turn on and off a given user's ability to discuss. However, usernames are relativly weak ways to control or exert power over the people who use those usernames. As O'Keefe explains, "The member may just sign up again, but the username is her identity on the site and should, as such, be the first thing you ban." There is nothing stopping this person from signing up for another account and starting to post again. Banning a username is not so much an exertion of technical control; people can sign up for accounts and start posting again. Banning a username is primarily about normative control. Publicly shaming someone and blocking them from participating in the community under a particular username and the identity that username represents.

From there O'Keefe discusses another kind of ban, banning IP addresses. He explains "Your community software should allow you to check what IPs a user has posted from." If it turns out that a user "made all of the posts from one address" banning their IP "may actually work." Even if they made most of their "posts from IPs that are similar except for the last few digits you can block an IP range and that also might work." With this noted he recognizes the substantive risks, "Besides not always being effective, it sometimes prevents other users on the same internet service provider (ISP) from reaching your community" [21]. While administrators exercise considerable control over their communities the tools at hand, blocking usernames and IP addresses, are both relatively blunt instruments. These methods are easily overcome by an antagonistic user. While banning an account is a trivial task, it doesn't actually stop someone from simply creating new accounts if they so desire. Similarly, one can get around an IP ban by using a proxy server. In a section titled "Get Creative" O'Keefe explains a series of ways to thwart attempts from bad actors to access and interact with an online community.

Creativity in this case tends to mean more levels of obfuscation and manipulation. First off this includes making it look like your site is down. "You could make it so that a 404 (not found) page displays when a specific IP visits your community" He explains how you can configure an .htaccess file to display a 403 (forbidden) page to any user from a particular IP range. Here, he is using Apache and the HTTP protocol to shut down participation. But beyond the protocol, he suggests adding an additional layer of deception. He explains, "You can customize your 403 page to look like a 404 page, which will give the impression that the site is down." [22] O'Keefe is not simply suggesting that one should use the HTTP protocol to block access, he wants you to take the additional step of misrepresenting what you are doing and making it seem like the site is down.

In case these ideas sound particularly extreme, so much so that you might think they are idiosyncratic to the author. O'Keefe goes on to explain some easy ways to "simulate downtime." In particular, the "Miserable Users" hack for vBulletin and "Troll" for phpBB., both of which "combine downtime, slowness, general confusion, and the site actually working." The goal of the "Troll" and "Miserable Users" hacks, like his suggestions for the 403 .htaccess hack is that they will hopefully "frustrate these troublemakers and drive them away." Not only are O'Keefe's ideas about how to deal with troublesome users more widespread, there was actually enough desire for such functionality that similar plugins were created for the two most popular discussion board software platforms.

Instead of making it look like the site itself is down O'Keefe offers a related approach for shutting particular people out of the conversation. "Sometimes referred to as global ignore, you can incorporate a function that lets the banned user log in but then makes their user go unseen to all users of your community." The user thinks they are participating in the community but they are not actually participating. "He can still make his posts, but only he (and maybe you and your staff) can see the posts--no one else. Basically, in his eyes, the site works as is intended. He will, hopefully, just think that everyone is ignoring him and go away." [23]

The globally ignored user has been muted, a rather deceptive practice. It is hard to conjure up a comparative situation in face-to-face communication. If you mute someone on a conference call, so that only they could hear themselves, it quickly becomes clear that no one is pausing, waiting to talk, or doing any of the other things we do when we are engaged in communication. Because the globally ignored individual continues to see themselves occupying the same kind of space in the threaded discussion it would likely take more time for them to realize what is going on in communication.

These deceptive practices illustrate a sophisticated mixture of extreme control and an extreme lack of control. The range of methods and approaches that an administrator can take are part of a complicated social dance, any and all of the technical approaches to banning and keeping users away come with significant limitations. An individual can simply sign up for a new account, or change their IP address. At the same time, it is clear from these extreme examples just how much power administrators have to shape and manipulate the experience of online discourse. The lessons for those interested in studying discourse and conversation in online communities should be the need for understanding the context of communication. There are clear practices of obfuscation that admins have at their disposal and there is every reason to believe that particular discussants who are deemed to be bad actors are being silenced in any number of online communities. This is not to suggest that there aren't ways of finding evidence and information about this silencing. In many cases, it would be valuable for a researcher to spend some time thinking about they ways users can be shut out of a particular online discussion and where the researcher might find traces or evidence either in the particular community or on other sites that explain how and why particular kinds of users were silenced.

Behaviorism Reputation and Social Hierarchy by Design

Anna Buss and Nancy Strauss's 2009 book, *Online Communities Handbook: Building Your Business and Brand on the Web* differs distinctly from the two books discussed thus far in its explicit commercial focus. Strauss and Buss both work as consultants with companies interested in web communities. While O'Keefe and Powazak discuss earning money, it is not the central frame through which they think about and describe online community. Published by New Riders Press, the *Online Communities Handbook* is similarly targeted at a technical audience but with much more of an explicit focus on how online communities can be used for marketing products. While discussing the same topics as the other books, here users are customers who are recruited, motivated, and, if they slow down their participation, reactivated.

The book offers many points of entry for understanding the values in the discourse of configuration in these how-to books; however, the chapter on motivation is particularly useful for exploring the interplay between a set of psycho-socio beliefs that underpin the authors' explanation of desired functionality in online community software. From the beginning, it is clear that not only is there a commercial goal for the sites, but that site users are explicitly thought of as commodities themselves. For example, when Buss and Strauss explain that the key question is "how can you get the maximum mileage from your members?" the community members explicitly become a fuel to be consumed [24]. Similarly, when they suggest, in a heading for a section on how to engage members, to "Grab them when they're fresh," members become some kind of perishable fruit or vegetable. In both cases users become something for you as the site administrator to consume. The authors go on to explain how this approach translates into specific functionality.

For Buss and Strauss, the first activity on your site needs to be simple, enjoyable, and show "an immediate result (for example, a photo appears on the page) for instant gratification" and promise a long-term result, like another member who might comment on it. The reference to "instant gratification" itself represents a longstanding connection between the discourse of marketing and the discourse of behaviorist psychology. It is unclear the extent to which this represents a rationalization of functionality on the terms of marketing or if this indicative of an ideology that has itself guided the development of these kinds of gratification cycles in the functionality of online community software. In any event, the link and relationship is worth further study and exploration.

At this point, Buss and Strauss discuss a marketing term that has become a common place term in eCommerce, the idea of stickiness. Buss and Struass describe stickiness as "website content that causes the user to spend more time on the site" [25] Stickiness is grounded in their pop-behaviorist psychology principles an explanation for how to run an online community. As in much of the book, the psychological ideas and the theory of society and community embedded in those notions are implied as statements of fact. These are simply statements about the way the world works which the authors translate into ways of understanding how to structure online community.

> Not only do you need users to spend time on the website, you need them to come back again and again. You should therefore aim for the right balance between instant and delayed gratification. On the one hand, users should see instant results from their actions, giving them a satisfying experience on the website. On the other hand, there should be benefits that build over time, enriching the experience the more time they spend in the community. [26]

Buss and Strauss are mobilizing a theory of human behavior, one that is focused on instant and delayed gratification, in service of particular designed features. The functionality they describe is a staple of the design of online community systems. They go on to explain how loyalty programs where "users earn points for each website action" and frequently remind them that a score of 200 points would let them access the site's video library can serve these self-gratifying desires. In the logic of their theory, these "virtual rewards" trigger the instant and delayed gratification. They remind us that these rewards should "depend on your target demographic," suggesting that "Teenagers are not interested in a business card exchange feature, while business users may be less likely to crave virtual pets to keep on their personal homepages." At each step, the theory of self becomes more unabashedly behaviorist. Reward the behavior you want, gratify the user, make and satisfy their cravings, keep pulling them back in and providing the stimulus of the reward. While Powazak and O'Keefe were clearly interested in manipulating users through various methods, they did not commodify users in the same way. Manipulation for Powazak and O'Keefe reads as primarily a paternalistic endeavor, trying to manipulate community members in the best interests of generating discussions that are valuable and meaningful to the community as a whole. For Buss and Strauss though, manipulation is a clear case of the exploitation of users as a means to a commercial end.

The features Buss and Strauss discuss have more recently been rebranded as "gameification" under the guise that they represent lessons from video game design applied to designing online communities. Clearly, the ideas of points, and badges and virtual rewards predate this conversation about gameification, suggesting that the new term is simply another guise under which the same behaviorist and consumptive notions of the self and society can be rebranded. With that said, Margaret Robertson's critique of gamification as "pointsification" and Ian Bogost's critique of it as "exploitation-ware" are equally well pointed critiques of this brand of thinking about users[27].

The behaviorist psychology of gratification and rewards becomes a theory of society as Buss and Strauss begin to explain the critical value of "social hierarchy" as an explanatory device for another particular set of functionality, the reputation system. They provide us with a theory of the social and then give us a case study from a site they worked on, Ciao.com, in which they illustrate what this theory looks like in practice. It is worth quoting these both at length to pick apart exactly what they are suggesting and what it looks like as a system.

> Social hierarchy in your community is a powerful tool. Just like offline communities, online ones quickly sort themselves into a hierarchical structure, normally with the most experienced members at the top. By encouraging hierarchy in your community and offering visible status symbols based on seniority and activity level, you can create an environment in which members feel as if they are working toward an objective: the next rung on the social ladder. [28]

For the most part, talk of tools in this essay so far has focused on software and plugins and hacks for software. Here, Buss and Strauss explain that one of our most powerful tools is not part of the software, it is social hierarchy and our innate desire for social hierarchy. When they assert the fact that online communities "quickly sort themselves into hierarchical structures" they naturalize the idea of hierarchy. Similarly, their assertion that this is "just like offline communities" further attempts to naturalize, and suggest the inevitability of such a situation. Halfway through the quote, they shift from asserting the natural-ness of social hierarchy to using that assertion to suggest online community administrators and managers should encourage this hierarchy to their ends. From there, they clearly evoke the same kind of stimulus reward ideas already espoused. A site admin will want to offer "visual status symbols" in this case of the characteristics they want to encourage, "seniority and activity level" and give users the feeling that they are "working toward an objective." Specifically, the members objective is to reach "the next rung on the social ladder," that is the instant and delayed gratification in this system is based explicitly on the idea that what motivates users (again the title of this chapter is motivation) is their desire to gain some arbitrary and non-monetary signifiers of their increasing social status in the given community site.

> The consumer community Ciao.com has a non-monetary rewards system that issues colorful dots as a status symbol. Members can earn points to change the color of their dot by posting product reviews that other members rate as useful, or by performing other community actions. You can find many members who post messages on their profile pages related to this community points systems. "Hooray, I'm finally red," they write. "Please read my product reviews and help me turn orange!" [29]

They start by describing Ciao.com as a "consumer community" and explicitly suggesting that the community uses a "non-monetary rewards system." Again, the very idea of a "rewards system" brings with it the behaviorist inclinations of gratification. At each step of the description we see ways in which users are trivialized. As users engage in the activities that Buss and Strauss have chosen to reinforce -- in this case writing product reviews and performing other undisclosed "community actions" -- they receive the points that enable them to change the color of their apparently arbitrarily and infinitely trivial "colored dots." They tell us of how their users exclaim "hooray" at finally being red, and attempt to recruit each other to read their product reviews to "help me turn orange!" In Buss and Strauss's presentation, users and community members are commodities which you consume and manipulate based on their desires for gratification. The very structure of social interaction enacted and prompted by the software's reputation system and usage of virtual rewards is explained as the result of projecting this behaviorist psychological model onto a view of the innate desire for "social ladder" climbing. The functionality of the software is explained to us as a manifestation of these asserted social and psychological facts.

Technical Literature, Configuring Discourse and Software

Powazak, O'Keefe, and Buss and Strauss's books each result from and articulate perspectives in the discourse of the design and implementation of the server side software which enables online community. The tactics each of these authors suggest explicate the goals and values of their work. These texts demonstarte the role software plays in online discourse and design elements that those interested in studying the textual record of online discourse should attend to in their studies.

Control, Empowerment and its Limits

Visual design, information architecture, text prompts and reward systems are all designed with the intention of stimulating particular types of discussion between particular kinds of users. Much of this is evident to someone who is willing to closely read the resulting interfaces. The placement of a post button, the structure of a reward system, all leave interpretable traces in the rendered pages on a user's screen. As is evident in O'Keefe's examples of extensive manipulation of problem users, administrators have considerable power to control what is passed to a given user's web browser. At the same time, the problems with different techniques for banning users illustrate the fundamentally limited nature of those control techniques. The technical means of control are limited by the ability of users to change their usernames, or IP address. The administrator's tools for banning a person are all tools for banning poor surrogates for people, usernames and IP addresses. The forum administrator depends on the norms and rules established in the posted textual guidelines for the site to establish and retain control.

Tthe nuances of experiences like O'Keefe's reported in these how-to books provides a potent vector for further triangulating the relationship between power and control on the web. The tension between control and empowerment found in O'Keefe's description provides nuanced validation of the same tensions Galloway found in his examination of the specifications for TCP/IP and DNS in *Protocols: How Control Exists After Decentralization*. In both cases, consideration of technical issues as documented in other texts, in Galloway's case in the specifications for the protocols that make the web work. and in this essay in the description of how to set up online community sites. Both offer substantive sources to explore how the constraints of the technical infrastructure of the web interact with human action to enable and disable particular instances of control and empowerment.

Theories of Users as Generalized Others in Design

These texts espouse theories of users. They each offer taxonomies of users. In this respect, the texts offer insight into ways administrators and developers think about their users. Exploring how these authors describe good and bad users -- whether as engaged and on topic participants and commentators or as the people who make them money -- offers a point of entry into how the generalized other of these user types play a role in the design and functionality of these systems. The various ideas of the good and bad user are sets of expactations through which functionality is described and explained.

These generalized others, these ideal types of users, play a key role in the design and configuration of online community software. The ideas of different kinds of users serve as warrants in the arguments that each of these authors present for why one should design and implement software in a particular way. Studying these texts suggests models for how developers' ideas of particular theoretical good and bad users play a role in their design decisions which in turn are manifest in the actual material affordances of the software itself.

Taking this idea seriously presents a potentially intriguing way to think about the discursive and social in software studies . The social and cultural is not to be understood as some outside force, but in keeping with Gee's approach to Discourse, any kind of social force is much more accurately thought of in terms of internalized theories about others. In this case, these are particularly interesting as each of these authors' theories of users and user behavior have been committed to the page and disseminated as cultural scripts for other developers and administrators to look up and potentially integrate into their internal theories of users. Beyond being inscribed on pages these ideas are encoded as the scripts by which the server side software we interact with as users operates. The ideas of users depicted in these texts become actants in our social interaction.

Software as Actant, Actor, and Mediator in Facework

Throughout the stories and advice in these texts, the functional and structural characteristics of these software systems can be thought of as interjecting themselves as actants, as mediators or as procedural participants in the facework that occurs in online communities. In the case of the excerpt from Powazak, the designer has interceded to structure the visual layout and interaction of the site to act as a filter for particular kinds of interaction. While he is not blocking out particular kinds of users he is explicitly suggesting that designers put together the infrastructure of online communities in ways that promote particular kinds of values. In the case of Buss and Strauss, we find suggestions from designers to use a behaviorist theory of the self and society to underpin a system of rewards and encouragements dolled out by the online community software to reinforce and reward particular behaviors that drive profit. From reading these texts we cannot know how individual users perceive or are affected by these mediators of interaction, but we do gain an understanding for how developers and administrators might be thinking about and designing for particular identified, intended behaviors.

The software is an actor, one that carries traces of the platform, of the plugins and hacks, of the administrator’s decisions. The scripts that pop up and give us helpful advice, the plugins that make it look like the site is down, the reputation system that tells me I have 200 points -- all are simultaneously manipulating me to behave in particular ways and enabling and empowering me to engage in specific behaviors. These how-to texts offer an approach for identifying how developers and administrators design for particular intended behavior and embed their values and ideas into the functionality of these systems. While designed with a particular set of theories and values about communication and users, the resulting software then acts with and upon users. Users interpret, work around and use the software to their ends and their use and action is itself interpreted and evaluated by the developers and admins. Developers, admins, users and software intermingle continually configuring and reconfiguring online discourse.

Notes

[1] For example, Black, *Adolescents and Online Fan Fiction*. suggests that participants are developing as writers and in some cases using these online communities to learn English as a second language. Similarly, studies of video game fan-forums suggest that young people are developing their abilities to interact with, critique and design video games. See, Duncan, “Gamers as Designers.” Squire and Giovanetto, “The Higher Education of Gaming.” and Owens, “Modding the History of Science.”

[2]Kozinets,. *Netnography: Doing Ethnographic Research Online*. Sage Publications Ltd,. 72

[3]Ibid, 68

[4]Miller & Slater. 2001. *The Internet: An Ethnographic Approach*, 14

[5] Famiglietti, Andrew. "The pentad of cruft: A taxonomy of rhetoric used by Wikipedia editors based on the dramatism of Kenneth Burke" *First Monday*

[6] Montfort & Bogost, Racing the Beam: The Atari Video Computer System.

[97] Gee, *An Introduction to Discourse Analysis: Theory and Method* and Fairclough, *Analysing Discourse: Textual Analysis for Social Research*.
[108] Gee, An Introduction to Discourse Analysis, xi\
[9] Goffman, Erving. *Interaction Ritual; Essays on Face-to-Face Behavior*.
[10] Buss and Strauss, *Online Communities Handbook*, 12
[11] Powazek, *Design for Community*, xxii.
[12] ibid, 53.
[13] ibid
[14] O'Keefe, *Managing Online Forums*, 185-199
[15] ibid, 204-205
[16] ibid, 206.
[17] ibid, 207.
[18] ibid, 25.
[19] ibid, 207.
[20] ibid.
[21] ibid, 212.
[22] ibid, 214.
[23] ibid
[24] Buss and Strauss, *Online Communities Handbook*, 80.
[25] ibid, 81.
[26] ibid, 83.
[27] Bogost, Ian. 2011. "Persuasive Games: Exploitationware" and Robertson, Margaret. 2010. "Can't Play, Won't Play"
[28] ibid, 86.
[29] ibid, 87

References:

Black, Rebecca W. 2008. *Adolescents and Online Fan Fiction*. New York: Peter Lang.

Benkler, Yochai. *The Wealth of Networks: How Social Production Transforms Markets and Freedom*. Yale University Press, 2006.

Bogost, Ian. 2011. "Persuasive Games: Exploitationware." http://www.gamasutra.com/view/feature/6366/persuasive_games_exploitationware.php.

Buss, Anna, and Nancy Strauss. 2009. *Online Communities Handbook: Building Your Business and Brand on the Web*. 1st ed. New Riders Press.

Duncan, Sean C. 2010. "Gamers as Designers: A Framework for Investigating Design in Gaming Affinity Spaces." *E-Learning and Digital Media* 7 (1): 21–34.

Fairclough, Norm. 2003. *Analysing Discourse: Textual Analysis for Social Research*. illustrated ed. Routledge.

Famiglietti, Andrew. 2012 "The pentad of cruft: A taxonomy of rhetoric used by Wikipedia editors based on the dramatism of Kenneth Burke" *First Monday* [Online], Volume 17 Number 9

Gee, James Paul. 2005. *An Introduction to Discourse Analysis: Theory and Method*. 2nd ed. London: Routledge.

Gibson, James. 1977. "The Theory of Affordances." In *Perceiving, Acting, and Knowing: Toward an Ecological Psychology*, ed. Robert Shaw and John Bransford. New York: Lawrence Erlbaum.

Goffman, Erving. *Interaction Ritual; Essays on Face-to-Face Behavior*. Garden City, N.Y: Anchor Books, 1967.

Kozinets, Robert V. 2009. *Netnography: Doing Ethnographic Research Online*. Sage Publications Ltd,.

Montfort, Nick, and Ian Bogost. 2009. *Racing the Beam: The Atari Video Computer System*. The MIT Press.

Miller, Daniel, and Don Slater. 2001. *The Internet: An Ethnographic Approach*. Berg Publishers.

O'Keefe, Patrick. 2008. *Managing Online Forums: Everything You Need to Know to Create and Run Successful Community Discussion Boards*. New York: AMACOM.

Owens, Trevor. 2010. "Modding the History of Science: Values at Play in Modder Discussions of Sid Meier's CIVILIZATION." *Simulation Gaming*: doi:10.1177/1046878110366277.

Powazek, Derek M. 2002. *Design for Community: The Art of Connecting Real People in Virtual Places*. Indianapolis, Ind: New Riders.

Robertson, Margaret. 2010. "Can't Play, Won't Play | Hide&Seek - Inventing New Kinds of Play." http://www.hideandseek.net/2010/10/06/cant-play-wont-play/.

Squire, Kurt, and L. Giovanetto. 2008. "The Higher Education of Gaming." *E-Learning* 5 (1): 2–28.