

The Wealth of Networks

How Social Production
Transforms Markets and
Freedom

Yochai Benkler

Yale University Press
New Haven and London

— I
— o
— + I

Copyright © 2006 by Yochai Benkler.
All rights reserved.

Subject to the exception immediately following, this book may not be reproduced, in whole or in part, including illustrations, in any form (beyond that copying permitted by Sections 107 and 108 of the U.S. Copyright Law and except by reviewers for the public press), without written permission from the publishers.

The author has made an online version of the book available under a Creative Commons Noncommercial Sharealike license; it can be accessed through the author's website at <http://www.benkler.org>.

Printed in the United States of America.

Library of Congress Cataloging-in-Publication Data

Benkler, Yochai.

The wealth of networks : how social production transforms markets and freedom / Yochai Benkler.

p. cm.

Includes bibliographical references and index.

ISBN-13: 978-0-300-11056-2 (alk. paper)

ISBN-10: 0-300-11056-1 (alk. paper)

1. Information society. 2. Information networks. 3. Computer networks—Social aspects. 4. Computer networks—Economic aspects.

I. Title.

HM851.B457 2006

303.48'33—dc22 2005028316

A catalogue record for this book is available from the British Library.

The paper in this book meets the guidelines for permanence and durability of the Committee on Production Guidelines for Book Longevity of the Council on Library Resources.

10 9 8 7 6 5 4 3 2 1

STRANGE FRUIT

By Lewis Allan

© 1939 (Renewed) by Music Sales Corporation (ASCAP)

International copyright secured. All rights reserved.

All rights outside the United States controlled by Edward B. Marks Music Company.

Reprinted by permission.

— I
— O
— + I

Chapter 3 Peer Production and Sharing

At the heart of the economic engine, of the world's most advanced economies, we are beginning to notice a persistent and quite amazing phenomenon. A new model of production has taken root; one that should not be there, at least according to our most widely held beliefs about economic behavior. It should not, the intuitions of the late-twentieth-century American would say, be the case that thousands of volunteers will come together to collaborate on a complex economic project. It certainly should not be that these volunteers will beat the largest and best-financed business enterprises in the world at their own game. And yet, this is precisely what is happening in the software world.

Industrial organization literature provides a prominent place for the transaction costs view of markets and firms, based on insights of Ronald Coase and Oliver Williamson. On this view, people use markets when the gains from doing so, net of transaction costs, exceed the gains from doing the same thing in a managed firm, net of the costs of organizing and managing a firm. Firms emerge when the opposite is true, and transaction costs can best be reduced by

— -I
— o
— +I

bringing an activity into a managed context that requires no individual transactions to allocate this resource or that effort. The emergence of free and open-source software, and the phenomenal success of its flagships, the GNU/Linux operating system, the Apache Web server, Perl, and many others, should cause us to take a second look at this dominant paradigm.¹ Free software projects do not rely on markets or on managerial hierarchies to organize production. Programmers do not generally participate in a project because someone who is their boss told them to, though some do. They do not generally participate in a project because someone offers them a price to do so, though some participants do focus on long-term appropriation through money-oriented activities, like consulting or service contracts. However, the critical mass of participation in projects cannot be explained by the direct presence of a price or even a future monetary return. This is particularly true of the all-important, microlevel decisions: who will work, with what software, on what project. In other words, programmers participate in free software projects without following the signals generated by market-based, firm-based, or hybrid models. In chapter 2 I focused on how the networked information economy departs from the industrial information economy by improving the efficacy of nonmarket production generally. Free software offers a glimpse at a more basic and radical challenge. It suggests that the networked environment makes possible a new modality of organizing production: radically decentralized, collaborative, and nonproprietary; based on sharing resources and outputs among widely distributed, loosely connected individuals who cooperate with each other without relying on either market signals or managerial commands. This is what I call “commons-based peer production.”

“Commons” refers to a particular institutional form of structuring the rights to access, use, and control resources. It is the opposite of “property” in the following sense: With property, law determines one particular person who has the authority to decide how the resource will be used. That person may sell it, or give it away, more or less as he or she pleases. “More or less” because property doesn’t mean anything goes. We cannot, for example, decide that we will give our property away to one branch of our family, as long as that branch has boys, and then if that branch has no boys, decree that the property will revert to some other branch of the family. That type of provision, once common in English property law, is now legally void for public policy reasons. There are many other things we cannot do with our property—like build on wetlands. However, the core characteristic of prop-

— I
— o
— +I

erty as the institutional foundation of markets is that the allocation of power to decide how a resource will be used is systematically and drastically asymmetric. That asymmetry permits the existence of “an owner” who can decide what to do, and with whom. We know that transactions must be made—rent, purchase, and so forth—if we want the resource to be put to some other use. The salient characteristic of commons, as opposed to property, is that no single person has exclusive control over the use and disposition of any particular resource in the commons. Instead, resources governed by commons may be used or disposed of by anyone among some (more or less well-defined) number of persons, under rules that may range from “anything goes” to quite crisply articulated formal rules that are effectively enforced.

Commons can be divided into four types based on two parameters. The first parameter is whether they are open to anyone or only to a defined group. The oceans, the air, and highway systems are clear examples of open commons. Various traditional pasture arrangements in Swiss villages or irrigation regions in Spain are now classic examples, described by Eleanor Ostrom, of limited-access common resources—where access is limited only to members of the village or association that collectively “owns” some defined pasturelands or irrigation system.² As Carol Rose noted, these are better thought of as limited common property regimes, rather than commons, because they behave as property vis-à-vis the entire world except members of the group who together hold them in common. The second parameter is whether a commons system is regulated or unregulated. Practically all well-studied, limited common property regimes are regulated by more or less elaborate rules—some formal, some social-conventional—governing the use of the resources. Open commons, on the other hand, vary widely. Some commons, called open access, are governed by no rule. Anyone can use resources within these types of commons at will and without payment. Air is such a resource, with respect to air intake (breathing, feeding a turbine). However, air is a regulated commons with regard to outtake. For individual human beings, breathing out is mildly regulated by social convention—you do not breath too heavily on another human being’s face unless forced to. Air is a more extensively regulated commons for industrial exhalation—in the shape of pollution controls. The most successful and obvious regulated commons in contemporary landscapes are the sidewalks, streets, roads, and highways that cover our land and regulate the material foundation of our ability to move from one place to the other. In all these cases, however, the characteristic of commons is that the constraints, if any, are symmetric

— I
— O
— +I

among all users, and cannot be unilaterally controlled by any single individual. The term “commons-based” is intended to underscore that what is characteristic of the cooperative enterprises I describe in this chapter is that they are not built around the asymmetric exclusion typical of property. Rather, the inputs and outputs of the process are shared, freely or conditionally, in an institutional form that leaves them equally available for all to use as they choose at their individual discretion. This latter characteristic—that commons leave individuals free to make their own choices with regard to resources managed as a commons—is at the foundation of the freedom they make possible. This is a freedom I return to in the discussion of autonomy. Not all commons-based production efforts qualify as peer production. Any production strategy that manages its inputs and outputs as commons locates that production modality outside the proprietary system, in a framework of social relations. It is the freedom to interact with resources and projects without seeking anyone’s permission that marks commons-based production generally, and it is also that freedom that underlies the particular efficiencies of peer production, which I explore in chapter 4.

The term “peer production” characterizes a subset of commons-based production practices. It refers to production systems that depend on individual action that is self-selected and decentralized, rather than hierarchically assigned. “Centralization” is a particular response to the problem of how to make the behavior of many individual agents cohere into an effective pattern or achieve an effective result. Its primary attribute is the separation of the locus of opportunities for action from the authority to choose the action that the agent will undertake. Government authorities, firm managers, teachers in a classroom, all occupy a context in which potentially many individual wills could lead to action, and reduce the number of people whose will is permitted to affect the actual behavior patterns that the agents will adopt. “Decentralization” describes conditions under which the actions of many agents cohere and are effective despite the fact that they do not rely on reducing the number of people whose will counts to direct effective action. A substantial literature in the past twenty years, typified, for example, by Charles Sabel’s work, has focused on the ways in which firms have tried to overcome the rigidities of managerial pyramids by decentralizing learning, planning, and execution of the firm’s functions in the hands of employees or teams. The most pervasive mode of “decentralization,” however, is the ideal market. Each individual agent acts according to his or her will. Coherence and efficacy emerge because individuals signal their wishes, and plan

— I
— o
— +I

their behavior not in cooperation with others, but by coordinating, understanding the will of others and expressing their own through the price system.

What we are seeing now is the emergence of more effective collective action practices that are decentralized but do not rely on either the price system or a managerial structure for coordination. In this, they complement the increasing salience of uncoordinated nonmarket behavior that we saw in chapter 2. The networked environment not only provides a more effective platform for action to nonprofit organizations that organize action like firms or to hobbyists who merely coexist coordinately. It also provides a platform for new mechanisms for widely dispersed agents to adopt radically decentralized cooperation strategies other than by using proprietary and contractual claims to elicit prices or impose managerial commands. This kind of information production by agents operating on a decentralized, nonproprietary model is not completely new. Science is built by many people contributing incrementally—not operating on market signals, not being handed their research marching orders by a boss—independently deciding what to research, bringing their collaboration together, and creating science. What we see in the networked information economy is a dramatic increase in the importance and the centrality of information produced in this way.

FREE/OPEN-SOURCE SOFTWARE

The quintessential instance of commons-based peer production has been free software. Free software, or open source, is an approach to software development that is based on shared effort on a nonproprietary model. It depends on many individuals contributing to a common project, with a variety of motivations, and sharing their respective contributions without any single person or entity asserting rights to exclude either from the contributed components or from the resulting whole. In order to avoid having the joint product appropriated by any single party, participants usually retain copyrights in their contribution, but license them to anyone—participant or stranger—on a model that combines a universal license to use the materials with licensing constraints that make it difficult, if not impossible, for any single contributor or third party to appropriate the project. This model of licensing is the most important institutional innovation of the free software movement. Its central instance is the GNU General Public License, or GPL.

— I
— o
— + I

64 The Networked Information Economy

This requires anyone who modifies software and distributes the modified version to license it under the same free terms as the original software. While there have been many arguments about how widely the provisions that prevent downstream appropriation should be used, the practical adoption patterns have been dominated by forms of licensing that prevent anyone from exclusively appropriating the contributions or the joint product. More than 85 percent of active free software projects include some version of the GPL or similarly structured license.³

Free software has played a critical role in the recognition of peer production, because software is a functional good with measurable qualities. It can be more or less authoritatively tested against its market-based competitors. And, in many instances, free software has prevailed. About 70 percent of Web server software, in particular for critical e-commerce sites, runs on the Apache Web server—free software.⁴ More than half of all back-office e-mail functions are run by one free software program or another. Google, Amazon, and CNN.com, for example, run their Web servers on the GNU/Linux operating system. They do this, presumably, because they believe this peer-produced operating system is more reliable than the alternatives, not because the system is “free.” It would be absurd to risk a higher rate of failure in their core business activities in order to save a few hundred thousand dollars on licensing fees. Companies like IBM and Hewlett Packard, consumer electronics manufacturers, as well as military and other mission-critical government agencies around the world have begun to adopt business and service strategies that rely and extend free software. They do this because it allows them to build better equipment, sell better services, or better fulfill their public role, even though they do not control the software development process and cannot claim proprietary rights of exclusion in the products of their contributions.

The story of free software begins in 1984, when Richard Stallman started working on a project of building a nonproprietary operating system he called GNU (GNU’s Not Unix). Stallman, then at the Massachusetts Institute of Technology (MIT), operated from political conviction. He wanted a world in which software enabled people to use information freely, where no one would have to ask permission to change the software they use to fit their needs or to share it with a friend for whom it would be helpful. These freedoms to share and to make your own software were fundamentally incompatible with a model of production that relies on property rights and markets, he thought, because in order for there to be a market in uses of

— I
— o
— + I

software, owners must be able to make the software unavailable to people who need it. These people would then pay the provider in exchange for access to the software or modification they need. If anyone can make software or share software they possess with friends, it becomes very difficult to write software on a business model that relies on excluding people from software they need unless they pay. As a practical matter, Stallman started writing software himself, and wrote a good bit of it. More fundamentally, he adopted a legal technique that started a snowball rolling. He could not write a whole operating system by himself. Instead, he released pieces of his code under a license that allowed anyone to copy, distribute, and modify the software in whatever way they pleased. He required only that, if the person who modified the software then distributed it to others, he or she do so under the exact same conditions that he had distributed his software. In this way, he invited all other programmers to collaborate with him on this development program, if they wanted to, on the condition that they be as generous with making their contributions available to others as he had been with his. Because he retained the copyright to the software he distributed, he could write this condition into the license that he attached to the software. This meant that anyone using or distributing the software as is, without modifying it, would not violate Stallman's license. They could also modify the software for their own use, and this would not violate the license. However, if they chose to distribute the modified software, they would violate Stallman's copyright unless they included a license identical to his with the software they distributed. This license became the GNU General Public License, or GPL. The legal jujitsu Stallman used—asserting his own copyright claims, but only to force all downstream users who wanted to rely on his contributions to make their own contributions available to everyone else—came to be known as “copyleft,” an ironic twist on copyright. This legal artifice allowed anyone to contribute to the GNU project without worrying that one day they would wake up and find that someone had locked them out of the system they had helped to build.

The next major step came when a person with a more practical, rather than prophetic, approach to his work began developing one central component of the operating system—the kernel. Linus Torvalds began to share the early implementations of his kernel, called Linux, with others, under the GPL. These others then modified, added, contributed, and shared among themselves these pieces of the operating system. Building on top of Stallman's foundation, Torvalds crystallized a model of production that was fun-

— I
— O
— + I

damentally different from those that preceded it. His model was based on voluntary contributions and ubiquitous, recursive sharing; on small incremental improvements to a project by widely dispersed people, some of whom contributed a lot, others a little. Based on our usual assumptions about volunteer projects and decentralized production processes that have no managers, this was a model that could not succeed. But it did.

It took almost a decade for the mainstream technology industry to recognize the value of free or open-source software development and its collaborative production methodology. As the process expanded and came to encompass more participants, and produce more of the basic tools of Internet connectivity—Web server, e-mail server, scripting—more of those who participated sought to “normalize” it, or, more specifically, to render it apolitical. Free software is about freedom (“free as in free speech, not free beer” is Stallman’s epitaph for it). “Open-source software” was chosen as a term that would not carry the political connotations. It was simply a mode of organizing software production that may be more effective than market-based production. This move to depoliticize peer production of software led to something of a schism between the free software movement and the communities of open source software developers. It is important to understand, however, that from the perspective of society at large and the historical trajectory of information production generally the abandonment of political motivation and the importation of free software into the mainstream have not made it less politically interesting, but more so. Open source and its wide adoption in the business and bureaucratic mainstream allowed free software to emerge from the fringes of the software world and move to the center of the public debate about practical alternatives to the current way of doing things.

So what is open-source software development? The best source for a phenomenology of open-source development continues to be Eric Raymond’s *Cathedral and Bazaar*, written in 1998. Imagine that one person, or a small group of friends, wants a utility. It could be a text editor, photo-retouching software, or an operating system. The person or small group starts by developing a part of this project, up to a point where the whole utility—if it is simple enough—or some important part of it, is functional, though it might have much room for improvement. At this point, the person makes the program freely available to others, with its source code—instructions in a human-readable language that explain how the software does whatever it does when compiled into a machine-readable language. When others begin

— I
— o
— +I

to use it, they may find bugs, or related utilities that they want to add (e.g., the photo-retouching software only increases size and sharpness, and one of its users wants it to allow changing colors as well). The person who has found the bug or is interested in how to add functions to the software may or may not be the best person in the world to actually write the software fix. Nevertheless, he reports the bug or the new need in an Internet forum of users of the software. That person, or someone else, then thinks that they have a way of tweaking the software to fix the bug or add the new utility. They then do so, just as the first person did, and release a new version of the software with the fix or the added utility. The result is a collaboration between three people—the first author, who wrote the initial software; the second person, who identified a problem or shortcoming; and the third person, who fixed it. This collaboration is not managed by anyone who organizes the three, but is instead the outcome of them all reading the same Internet-based forum and using the same software, which is released under an open, rather than proprietary, license. This enables some of its users to identify problems and others to fix these problems without asking anyone's permission and without engaging in any transactions.

The most surprising thing that the open source movement has shown, in real life, is that this simple model can operate on very different scales, from the small, three-person model I described for simple projects, up to the many thousands of people involved in writing the Linux kernel and the GNU/Linux operating system—an immensely difficult production task. SourceForge, the most popular hosting-meeting place of such projects, has close to 100,000 registered projects, and nearly a million registered users. The economics of this phenomenon are complex. In the larger-scale models, actual organization form is more diverse than the simple, three-person model. In particular, in some of the larger projects, most prominently the Linux kernel development process, a certain kind of meritocratic hierarchy is clearly present. However, it is a hierarchy that is very different in style, practical implementation, and organizational role than that of the manager in the firm. I explain this in chapter 4, as part of the analysis of the organizational forms of peer production. For now, all we need is a broad outline of how peer-production projects look, as we turn to observe case studies of kindred production models in areas outside of software.

— I
— O
— +I

**PEER PRODUCTION OF INFORMATION,
KNOWLEDGE, AND CULTURE GENERALLY**

Free software is, without a doubt, the most visible instance of peer production at the turn of the twenty-first century. It is by no means, however, the only instance. Ubiquitous computer communications networks are bringing about a dramatic change in the scope, scale, and efficacy of peer production throughout the information and cultural production system. As computers become cheaper and as network connections become faster, cheaper, and ubiquitous, we are seeing the phenomenon of peer production of information scale to much larger sizes, performing more complex tasks than were possible in the past for nonprofessional production. To make this phenomenon more tangible, I describe a number of such enterprises, organized to demonstrate the feasibility of this approach throughout the information production and exchange chain. While it is possible to break an act of communication into finer-grained subcomponents, largely we see three distinct functions involved in the process. First, there is an initial utterance of a humanly meaningful statement. Writing an article or drawing a picture, whether done by a professional or an amateur, whether high quality or low, is such an action. Second, there is a separate function of mapping the initial utterances on a knowledge map. In particular, an utterance must be understood as “relevant” in some sense, and “credible.” Relevance is a subjective question of mapping an utterance on the conceptual map of a given user seeking information for a particular purpose defined by that individual. Credibility is a question of quality by some objective measure that the individual adopts as appropriate for purposes of evaluating a given utterance. The distinction between the two is somewhat artificial, however, because very often the utility of a piece of information will depend on a combined valuation of its credibility and relevance. I therefore refer to “relevance/accreditation” as a single function for purposes of this discussion, keeping in mind that the two are complementary and not entirely separable functions that an individual requires as part of being able to use utterances that others have uttered in putting together the user’s understanding of the world. Finally, there is the function of distribution, or how one takes an utterance produced by one person and distributes it to other people who find it credible and relevant. In the mass-media world, these functions were often, though by no means always, integrated. NBC news produced the utterances, gave them credibility by clearing them on the evening news, and distributed

— I
— O
— +I

them simultaneously. What the Internet is permitting is much greater disaggregation of these functions.

Uttering Content

NASA Clickworkers was “an experiment to see if public volunteers, each working for a few minutes here and there can do some routine science analysis that would normally be done by a scientist or graduate student working for months on end.” Users could mark craters on maps of Mars, classify craters that have already been marked, or search the Mars landscape for “honeycomb” terrain. The project was “a pilot study with limited funding, run part-time by one software engineer, with occasional input from two scientists.” In its first six months of operation, more than 85,000 users visited the site, with many contributing to the effort, making more than 1.9 million entries (including redundant entries of the same craters, used to average out errors). An analysis of the quality of markings showed “that the automatically-computed consensus of a large number of clickworkers is virtually indistinguishable from the inputs of a geologist with years of experience in identifying Mars craters.”⁵ The tasks performed by clickworkers (like marking craters) were discrete, each easily performed in a matter of minutes. As a result, users could choose to work for a few minutes doing a single iteration or for hours by doing many. An early study of the project suggested that some clickworkers indeed worked on the project for weeks, but that 37 percent of the work was done by one-time contributors.⁶

The clickworkers project was a particularly clear example of how a complex professional task that requires a number of highly trained individuals on full-time salaries can be reorganized so as to be performed by tens of thousands of volunteers in increments so minute that the tasks could be performed on a much lower budget. The low budget would be devoted to coordinating the volunteer effort. However, the raw human capital needed would be contributed for the fun of it. The professionalism of the original scientists was replaced by a combination of high modularization of the task. The organizers broke a large, complex task into small, independent modules. They built in redundancy and automated averaging out of both errors and purposeful erroneous markings—like those of an errant art student who thought it amusing to mark concentric circles on the map. What the NASA scientists running this experiment had tapped into was a vast pool of five-minute increments of human judgment, applied with motivation to participate in a task unrelated to “making a living.”

— I
— O
— +I

While clickworkers was a distinct, self-conscious experiment, it suggests characteristics of distributed production that are, in fact, quite widely observable. We have already seen in chapter 2, in our little search for Viking ships, how the Internet can produce encyclopedic or almanac-type information. The power of the Web to answer such an encyclopedic question comes not from the fact that one particular site has all the great answers. It is not an *Encyclopædia Britannica*. The power comes from the fact that it allows a user looking for specific information at a given time to collect answers from a sufficiently large number of contributions. The task of sifting and accrediting falls to the user, motivated by the need to find an answer to the question posed. As long as there are tools to lower the cost of that task to a level acceptable to the user, the Web shall have “produced” the information content the user was looking for. These are not trivial considerations, but they are also not intractable. As we shall see, some of the solutions can themselves be peer produced, and some solutions are emerging as a function of the speed of computation and communication, which enables more efficient technological solutions.

Encyclopedic and almanac-type information emerges on the Web out of the coordinate but entirely independent action of millions of users. This type of information also provides the focus on one of the most successful collaborative enterprises that has developed in the first five years of the twenty-first century, *Wikipedia*. *Wikipedia* was founded by an Internet entrepreneur, Jimmy Wales. Wales had earlier tried to organize an encyclopedia named Nupedia, which was built on a traditional production model, but whose outputs were to be released freely: its contributors were to be PhDs, using a formal, peer-reviewed process. That project appears to have failed to generate a sufficient number of high-quality contributions, but its outputs were used in *Wikipedia* as the seeds for a radically new form of encyclopedia writing. Founded in January 2001, *Wikipedia* combines three core characteristics: First, it uses a collaborative authorship tool, Wiki. This platform enables anyone, including anonymous passersby, to edit almost any page in the entire project. It stores all versions, makes changes easily visible, and enables anyone to revert a document to any prior version as well as to add changes, small and large. All contributions and changes are rendered transparent by the software and database. Second, it is a self-conscious effort at creating an encyclopedia—governed first and foremost by a collective informal undertaking to strive for a neutral point of view, within the limits of substantial self-awareness as to the difficulties of such an enterprise. An effort

— I
— O
— + I

to represent sympathetically all views on a subject, rather than to achieve objectivity, is the core operative characteristic of this effort. Third, all the content generated by this collaboration is released under the GNU Free Documentation License, an adaptation of the GNU GPL to texts.

The shift in strategy toward an open, peer-produced model proved enormously successful. The site saw tremendous growth both in the number of contributors, including the number of active and very active contributors, and in the number of articles included in the encyclopedia (table 3.1). Most of the early growth was in English, but more recently there has been an increase in the number of articles in many other languages: most notably in German (more than 200,000 articles), Japanese (more than 120,000 articles), and French (about 100,000), but also in another five languages that have between 40,000 and 70,000 articles each, another eleven languages with 10,000 to 40,000 articles each, and thirty-five languages with between 1,000 and 10,000 articles each.

The first systematic study of the quality of *Wikipedia* articles was published as this book was going to press. The journal *Nature* compared 42 science articles from *Wikipedia* to the gold standard of the *Encyclopedia Britannica*, and concluded that “the difference in accuracy was not particularly great.”⁷ On November 15, 2004, Robert McHenry, a former editor in chief of the *Encyclopedia Britannica*, published an article criticizing *Wikipedia* as “The Faith-Based Encyclopedia.”⁸ As an example, McHenry mocked the *Wikipedia* article on Alexander Hamilton. He noted that Hamilton biographers have a problem fixing his birth year—whether it is 1755 or 1757. *Wikipedia* glossed over this error, fixing the date at 1755. McHenry then went on to criticize the way the dates were treated throughout the article, using it as an anchor to his general claim: *Wikipedia* is unreliable because it is not professionally produced. What McHenry did not note was that the other major online encyclopedias—like *Columbia* or *Encarta*—similarly failed to deal with the ambiguity surrounding Hamilton’s birth date. Only the *Britannica* did. However, McHenry’s critique triggered the *Wikipedia* distributed correction mechanism. Within hours of the publication of McHenry’s Web article, the reference was corrected. The following few days saw intensive cleanup efforts to conform all references in the biography to the newly corrected version. Within a week or so, *Wikipedia* had a correct, reasonably clean version. It now stood alone with the *Encyclopedia Britannica* as a source of accurate basic encyclopedic information. In coming to curse it, McHenry found himself blessing *Wikipedia*. He had demonstrated

— I
 — O
 — + I

Table 3.1: Contributors to *Wikipedia*, January 2001–June 2005

	Jan. 2001	Jan. 2002	Jan. 2003	Jan. 2004	July 2004	June 2005
Contributors*	10	472	2,188	9,653	25,011	48,721
Active contributors**	9	212	846	3,228	8,442	16,945
Very active contributors***	0	31	190	692	1,637	3,016
No. of English language articles	25	16,000	101,000	190,000	320,000	630,000
No. of articles, all languages	25	19,000	138,000	409,000	862,000	1,600,000

* Contributed at least ten times; ** at least 5 times in last month; *** more than 100 times in last month.

precisely the correction mechanism that makes *Wikipedia*, in the long term, a robust model of reasonably reliable information.

Perhaps the most interesting characteristic about *Wikipedia* is the self-conscious social-norms-based dedication to objective writing. Unlike some of the other projects that I describe in this chapter, *Wikipedia* does not include elaborate software-controlled access and editing capabilities. It is generally open for anyone to edit the materials, delete another's change, debate the desirable contents, survey archives for prior changes, and so forth. It depends on self-conscious use of open discourse, usually aimed at consensus. While there is the possibility that a user will call for a vote of the participants on any given definition, such calls can, and usually are, ignored by the community unless a sufficiently large number of users have decided that debate has been exhausted. While the system operators and server host—Wales—have the practical power to block users who are systematically disruptive, this power seems to be used rarely. The project relies instead on social norms to secure the dedication of project participants to objective writing. So, while not entirely anarchic, the project is nonetheless substantially more social, human, and intensively discourse- and trust-based than the other major projects described here. The following fragments from an early version of the self-described essential characteristics and basic policies of *Wikipedia* are illustrative:

First and foremost, the Wikipedia project is self-consciously an encyclopedia—rather than a dictionary, discussion forum, web portal, etc. Wikipedia's partici-

— I
— O
— +I

pants commonly follow, and enforce, a few basic policies that seem essential to keeping the project running smoothly and productively. First, because we have a huge variety of participants of all ideologies, and from around the world, Wikipedia is committed to making its articles as unbiased as possible. The aim is not to write articles from a single *objective* point of view—this is a common misunderstanding of the policy—but rather, to fairly and sympathetically present all views on an issue. See “neutral point of view” page for further explanation.⁹

The point to see from this quotation is that the participants of *Wikipedia* are plainly people who like to write. Some of them participate in other collaborative authorship projects. However, when they enter the common project of *Wikipedia*, they undertake to participate in a particular way—a way that the group has adopted to make its product be an encyclopedia. On their interpretation, that means conveying in brief terms the state of the art on the item, including divergent opinions about it, but not the author’s opinion. Whether that is an attainable goal is a subject of interpretive theory, and is a question as applicable to a professional encyclopedia as it is to *Wikipedia*. As the project has grown, it has developed more elaborate spaces for discussing governance and for conflict resolution. It has developed structures for mediation, and if that fails, arbitration, of disputes about particular articles.

The important point is that *Wikipedia* requires not only mechanical cooperation among people, but a commitment to a particular style of writing and describing concepts that is far from intuitive or natural to people. It requires self-discipline. It enforces the behavior it requires primarily through appeal to the common enterprise that the participants are engaged in, coupled with a thoroughly transparent platform that faithfully records and renders all individual interventions in the common project and facilitates discourse among participants about how their contributions do, or do not, contribute to this common enterprise. This combination of an explicit statement of common purpose, transparency, and the ability of participants to identify each other’s actions and counteract them—that is, edit out “bad” or “faithless” definitions—seems to have succeeded in keeping this community from devolving into inefficacy or worse. A case study by IBM showed, for example, that while there were many instances of vandalism on *Wikipedia*, including deletion of entire versions of articles on controversial topics like “abortion,” the ability of users to see what was done and to fix it with a single click by reverting to a past version meant that acts of vandalism were

— I
— O
— +I

corrected within minutes. Indeed, corrections were so rapid that vandalism acts and their corrections did not even appear on a mechanically generated image of the abortion definition as it changed over time.¹⁰ What is perhaps surprising is that this success occurs not in a tightly knit community with many social relations to reinforce the sense of common purpose and the social norms embodying it, but in a large and geographically dispersed group of otherwise unrelated participants. It suggests that even in a group of this size, social norms coupled with a facility to allow any participant to edit out purposeful or mistaken deviations in contravention of the social norms, and a robust platform for largely unmediated conversation, keep the group on track.

A very different cultural form of distributed content production is presented by the rise of massive multiplayer online games (MMOGs) as immersive entertainment. These fall in the same cultural “time slot” as television shows and movies of the twentieth century. The interesting thing about these types of games is that they organize the production of “scripts” very differently from movies or television shows. In a game like *Ultima Online* or *EverQuest*, the role of the commercial provider is not to tell a finished, highly polished story to be consumed start to finish by passive consumers. Rather, the role of the game provider is to build tools with which users collaborate to tell a story. There have been observations about this approach for years, regarding MUDs (Multi-User Dungeons) and MOOs (Multi-User Object Oriented games). The point to understand about MMOGs is that they produce a discrete element of “content” that was in the past dominated by centralized professional production. The screenwriter of an immersive entertainment product like a movie is like the scientist marking Mars craters—a professional producer of a finished good. In MMOGs, this function is produced by using the appropriate software platform to allow the story to be written by the many users as they experience it. The individual contributions of the users/coauthors of the story line are literally done for fun—they are playing a game. However, they are spending real economic goods—their attention and substantial subscription fees—on a form of entertainment that uses a platform for active coproduction of a story line to displace what was once passive reception of a finished, commercially and professionally manufactured good.

By 2003, a company called Linden Lab took this concept a major step forward by building an online game environment called *Second Life*. *Second Life* began almost entirely devoid of content. It was tools all the way down.

— I
— o
— + I

Within a matter of months, it had thousands of subscribers, inhabiting a “world” that had thousands of characters, hundreds of thousands of objects, multiple areas, villages, and “story lines.” The individual users themselves had created more than 99 percent of all objects in the game environment, and all story lines and substantive frameworks for interaction—such as a particular village or group of theme-based participants. The interactions in the game environment involved a good deal of gift giving and a good deal of trade, but also some very surprising structured behaviors. Some users set up a university, where lessons were given in both in-game skills and in programming. Others designed spaceships and engaged in alien abductions (undergoing one seemed to become a status symbol within the game). At one point, aiming (successfully) to prevent the company from changing its pricing policy, users staged a demonstration by making signs and picketing the entry point to the game; and a “tax revolt” by placing large numbers of “tea crates” around an in-game reproduction of the Washington Monument. Within months, Second Life had become an immersive experience, like a movie or book, but one where the commercial provider offered a platform and tools, while the users wrote the story lines, rendered the “set,” and performed the entire play.

Relevance/Accreditation

How are we to know that the content produced by widely dispersed individuals is not sheer gobbledygook? Can relevance and accreditation itself be produced on a peer-production model? One type of answer is provided by looking at commercial businesses that successfully break off precisely the “accreditation and relevance” piece of their product, and rely on peer production to perform that function. Amazon and Google are probably the two most prominent examples of this strategy.

Amazon uses a mix of mechanisms to get in front of their buyers of books and other products that the users are likely to purchase. A number of these mechanisms produce relevance and accreditation by harnessing the users themselves. At the simplest level, the recommendation “customers who bought items you recently viewed also bought these items” is a mechanical means of extracting judgments of relevance and accreditation from the actions of many individuals, who produce the datum of relevance as by-product of making their own purchasing decisions. Amazon also allows users to create topical lists and track other users as their “friends and favorites.” Amazon, like many consumer sites today, also provides users with the ability

— I
— O
— +I

to rate books they buy, generating a peer-produced rating by averaging the ratings. More fundamentally, the core innovation of Google, widely recognized as the most efficient general search engine during the first half of the 2000s, was to introduce peer-based judgments of relevance. Like other search engines at the time, Google used a text-based algorithm to retrieve a given universe of Web pages initially. Its major innovation was its PageRank algorithm, which harnesses peer production of ranking in the following way. The engine treats links from other Web sites pointing to a given Web site as votes of confidence. Whenever someone who authors a Web site links to someone else's page, that person has stated quite explicitly that the linked page is worth a visit. Google's search engine counts these links as distributed votes of confidence in the quality of the page pointed to. Pages that are heavily linked-to count as more important votes of confidence. If a highly linked-to site links to a given page, that vote counts for more than the vote of a site that no one else thinks is worth visiting. The point to take home from looking at Google and Amazon is that corporations that have done immensely well at acquiring and retaining users have harnessed peer production to enable users to find things they want quickly and efficiently.

The most prominent example of a distributed project self-consciously devoted to peer production of relevance is the Open Directory Project. The site relies on more than sixty thousand volunteer editors to determine which links should be included in the directory. Acceptance as a volunteer requires application. Quality relies on a peer-review process based substantially on seniority as a volunteer and level of engagement with the site. The site is hosted and administered by Netscape, which pays for server space and a small number of employees to administer the site and set up the initial guidelines. Licensing is free and presumably adds value partly to America Online's (AOL's) and Netscape's commercial search engine/portal and partly through goodwill. Volunteers are not affiliated with Netscape and receive no compensation. They spend time selecting sites for inclusion in the directory (in small increments of perhaps fifteen minutes per site reviewed), producing the most comprehensive, highest-quality human-edited directory of the Web—at this point outshining the directory produced by the company that pioneered human edited directories of the Web: Yahoo!.

Perhaps the most elaborate platform for peer production of relevance and accreditation, at multiple layers, is used by Slashdot. Billed as "News for Nerds," Slashdot has become a leading technology newsletter on the Web, coproduced by hundreds of thousands of users. Slashdot primarily consists

— I
— o
— + I

of users commenting on initial submissions that cover a variety of technology-related topics. The submissions are typically a link to an off-site story, coupled with commentary from the person who submits the piece. Users follow up the initial submission with comments that often number in the hundreds. The initial submissions themselves, and more importantly, the approach to sifting through the comments of users for relevance and accreditation, provide a rich example of how this function can be performed on a distributed, peer-production model.

First, it is important to understand that the function of posting a story from another site onto Slashdot, the first “utterance” in a chain of comments on Slashdot, is itself an act of relevance production. The person submitting the story is telling the community of Slashdot users, “here is a story that ‘News for Nerds’ readers should be interested in.” This initial submission of a link is itself very coarsely filtered by editors who are paid employees of Open Source Technology Group (OSTG), which runs a number of similar platforms—like SourceForge, the most important platform for free software developers. OSTG is a subsidiary of VA Software, a software services company. The FAQ (Frequently Asked Question) response to, “how do you verify the accuracy of Slashdot stories?” is revealing: “We don’t. You do. If something seems outrageous, we might look for some corroboration, but as a rule, we regard this as the responsibility of the submitter and the audience. This is why it’s important to read comments. You might find something that refutes, or supports, the story in the main.” In other words, Slashdot very self-consciously is organized as a means of facilitating peer production of accreditation; it is at the comments stage that the story undergoes its most important form of accreditation—peer review *ex-post*.

Filtering and accreditation of comments on Slashdot offer the most interesting case study of peer production of these functions. Users submit comments that are displayed together with the initial submission of a story. Think of the “content” produced in these comments as a cross between academic peer review of journal submissions and a peer-produced substitute for television’s “talking heads.” It is in the means of accrediting and evaluating these comments that Slashdot’s system provides a comprehensive example of peer production of relevance and accreditation. Slashdot implements an automated system to select moderators from the pool of users. Moderators are chosen according to several criteria; they must be logged in (not anonymous), they must be regular users (who use the site averagely, not one-time page loaders or compulsive users), they must have been using

— —I
— — O
— —+I

the site for a while (this defeats people who try to sign up just to moderate), they must be willing, and they must have positive “karma.” Karma is a number assigned to a user that primarily reflects whether he or she has posted good or bad comments (according to ratings from other moderators). If a user meets these criteria, the program assigns the user moderator status and the user gets five “influence points” to review comments. The moderator rates a comment of his choice using a drop-down list with words such as “flamebait” and “informative.” A positive word increases the rating of a comment one point and a negative word decreases the rating a point. Each time a moderator rates a comment, it costs one influence point, so he or she can only rate five comments for each moderating period. The period lasts for three days and if the user does not use the influence points, they expire. The moderation setup is designed to give many users a small amount of power. This decreases the effect of users with an ax to grind or with poor judgment. The site also implements some automated “troll filters,” which prevent users from sabotaging the system. Troll filters stop users from posting more than once every sixty seconds, prevent identical posts, and will ban a user for twenty-four hours if he or she has been moderated down several times within a short time frame. Slashdot then provides users with a “threshold” filter that allows each user to block lower-quality comments. The scheme uses the numerical rating of the comment (ranging from -1 to 5). Comments start out at 0 for anonymous posters, 1 for registered users, and 2 for registered users with good “karma.” As a result, if a user sets his or her filter at 1 , the user will not see any comments from anonymous posters unless the comments’ ratings were increased by a moderator. A user can set his or her filter anywhere from -1 (viewing all of the comments) to 5 (where only the posts that have been upgraded by several moderators will show up).

Relevance, as distinct from accreditation, is also tied into the Slashdot scheme because off-topic posts should receive an “off topic” rating by the moderators and sink below the threshold level (assuming the user has the threshold set above the minimum). However, the moderation system is limited to choices that sometimes are not mutually exclusive. For instance, a moderator may have to choose between “funny” ($+1$) and “off topic” (-1) when a post is both funny and off topic. As a result, an irrelevant post can increase in ranking and rise above the threshold level because it is funny or informative. It is unclear, however, whether this is a limitation on relevance, or indeed mimics our own normal behavior, say in reading a newspaper or browsing a library, where we might let our eyes linger longer on a funny or

— -1
 — 0
 — $+1$

informative tidbit, even after we have ascertained that it is not exactly relevant to what we were looking for.

The primary function of moderation is to provide accreditation. If a user sets a high threshold level, they will only see posts that are considered of high quality by the moderators. Users also receive accreditation through their karma. If their posts consistently receive high ratings, their karma will increase. At a certain karma level, their comments will start off with a rating of 2, thereby giving them a louder voice in the sense that users with a threshold of 2 will now see their posts immediately, and fewer upward moderations are needed to push their comments even higher. Conversely, a user with bad karma from consistently poorly rated comments can lose accreditation by having his or her posts initially start off at 0 or -1 . In addition to the mechanized means of selecting moderators and minimizing their power to skew the accreditation system, Slashdot implements a system of peer-review accreditation for the moderators themselves. Slashdot accomplishes this “metamoderation” by making any user that has an account from the first 90 percent of accounts created on the system eligible to evaluate the moderators. Each eligible user who opts to perform metamoderation review is provided with ten random moderator ratings of comments. The user/metamoderator then rates the moderator’s rating as either unfair, fair, or neither. The metamoderation process affects the karma of the original moderator, which, when lowered sufficiently by cumulative judgments of unfair ratings, will remove the moderator from the moderation system.

Together, these mechanisms allow for distributed production of both relevance and accreditation. Because there are many moderators who can moderate any given comment, and thanks to the mechanisms that explicitly limit the power of any one moderator to overinfluence the aggregate judgment, the system evens out differences in evaluation by aggregating judgments. It then allows individual users to determine what level of accreditation pronounced by this aggregate system fits their particular time and needs by setting their filter to be more or less inclusive. By introducing “karma,” the system also allows users to build reputation over time, and to gain greater control over the accreditation of their own work relative to the power of the critics. Users, moderators, and metamoderators are all volunteers.

The primary point to take from the Slashdot example is that the same dynamic that we saw used for peer production of initial utterances, or content, can be implemented to produce relevance and accreditation. Rather than using the full-time effort of professional accreditation experts, the sys-

— -1
— 0
— $+1$

tem is designed to permit the aggregation of many small judgments, each of which entails a trivial effort for the contributor, regarding both relevance and accreditation of the materials. The software that mediates the communication among the collaborating peers embeds both the means to facilitate the participation and a variety of mechanisms designed to defend the common effort from poor judgment or defection.

Value-Added Distribution

Finally, when we speak of information or cultural goods that exist (content has been produced) and are made usable through some relevance and accreditation mechanisms, there remains the question of distribution. To some extent, this is a nonissue on the Internet. Distribution is cheap. All one needs is a server and large pipes connecting one's server to the world. Nonetheless, this segment of the publication process has also provided us with important examples of peer production, including one of its earliest examples—Project Gutenberg.

Project Gutenberg entails hundreds of volunteers who scan in and correct books so that they are freely available in digital form. It has amassed more than 13,000 books, and makes the collection available to everyone for free. The vast majority of the “e-texts” offered are public domain materials. The site itself presents the e-texts in ASCII format, the lowest technical common denominator, but does not discourage volunteers from offering the e-texts in markup languages. It contains a search engine that allows a reader to search for typical fields such as subject, author, and title. Project Gutenberg volunteers can select any book that is in the public domain to transform into an e-text. The volunteer submits a copy of the title page of the book to Michael Hart—who founded the project—for copyright research. The volunteer is notified to proceed if the book passes the copyright clearance. The decision on which book to convert to e-text is left up to the volunteer, subject to copyright limitations. Typically, a volunteer converts a book to ASCII format using OCR (optical character recognition) and proofreads it one time in order to screen it for major errors. He or she then passes the ASCII file to a volunteer proofreader. This exchange is orchestrated with very little supervision. The volunteers use a Listserv mailing list and a bulletin board to initiate and supervise the exchange. In addition, books are labeled with a version number indicating how many times they have been proofed. The site encourages volunteers to select a book that has a low number and proof it. The Project Gutenberg proofing process is simple.

— I
— o
— +I

Proofreaders (aside from the first pass) are not expected to have access to the book, but merely review the e-text for self-evident errors.

Distributed Proofreading, a site originally unaffiliated with Project Gutenberg, is devoted to proofing Project Gutenberg e-texts more efficiently, by distributing the volunteer proofreading function in smaller and more information-rich modules. Charles Franks, a computer programmer from Las Vegas, decided that he had a more efficient way to proofread these e-texts. He built an interface that allowed volunteers to compare scanned images of original texts with the e-texts available on Project Gutenberg. In the Distributed Proofreading process, scanned pages are stored on the site, and volunteers are shown a scanned page and a page of the e-text simultaneously so that they can compare the e-text to the original page. Because of the fine-grained modularity, proofreaders can come on the site and proof one or a few pages and submit them. By contrast, on the Project Gutenberg site, the entire book is typically exchanged, or at minimum, a chapter. In this fashion, Distributed Proofreading clears the proofing of tens of thousands of pages every month. After a couple of years of working independently, Franks joined forces with Hart. By late 2004, the site had proofread more than five thousand volumes using this method.

Sharing of Processing, Storage, and Communications Platforms

All the examples of peer production that we have seen up to this point have been examples where individuals pool their time, experience, wisdom, and creativity to form new information, knowledge, and cultural goods. As we look around the Internet, however, we find that users also cooperate in similar loosely affiliated groups, without market signals or managerial commands, to build supercomputers and massive data storage and retrieval systems. In their radical decentralization and reliance on social relations and motivations, these sharing practices are similar to peer production of information, knowledge, and culture. They differ in one important aspect: Users are not sharing their innate and acquired human capabilities, and, unlike information, their inputs and outputs are not public goods. The participants are, instead, sharing material goods that they privately own, mostly personal computers and their components. They produce economic, not public, goods—computation, storage, and communications capacity.

As of the middle of 2004, the fastest supercomputer in the world was SETI@home. It ran about 75 percent faster than the supercomputer that

— I
— O
— + I

was then formally known as “the fastest supercomputer in the world”: the IBM Blue Gene/L. And yet, there was and is no single SETI@home computer. Instead, the SETI@home project has developed software and a collaboration platform that have enabled millions of participants to pool their computation resources into a single powerful computer. Every user who participates in the project must download a small screen saver. When a user’s personal computer is idle, the screen saver starts up, downloads problems for calculation—in SETI@home, these are radio astronomy signals to be analyzed for regularities—and calculates the problem it has downloaded. Once the program calculates a solution, it automatically sends its results to the main site. The cycle continues for as long as, and repeats every time that, the computer is idle from its user’s perspective. As of the middle of 2004, the project had harnessed the computers of 4.5 million users, allowing it to run computations at speeds greater than those achieved by the fastest supercomputers in the world that private firms, using full-time engineers, developed for the largest and best-funded government laboratories in the world. SETI@home is the most prominent, but is only one among dozens of similarly structured Internet-based distributed computing platforms. Another, whose structure has been the subject of the most extensive formal analysis by its creators, is Folding@home. As of mid-2004, Folding@home had amassed contributions of about 840,000 processors contributed by more than 365,000 users.

SETI@home and Folding@home provide a good basis for describing the fairly common characteristics of Internet-based distributed computation projects. First, these are noncommercial projects, engaged in pursuits understood as scientific, for the general good, seeking to harness contributions of individuals who wish to contribute to such larger-than-themselves goals. SETI@home helps in the search for extraterrestrial intelligence. Folding@home helps in protein folding research. Fightaids@home is dedicated to running models that screen compounds for the likelihood that they will provide good drug candidates to fight HIV/AIDS. Genome@home is dedicated to modeling artificial genes that would be created to generate useful proteins. Other sites, like those dedicated to cryptography or mathematics, have a narrower appeal, and combine “altruistic” with hobby as their basic motivational appeal. The absence of money is, in any event, typical of the large majority of active distributed computing projects. Less than one-fifth of these projects mention money at all. Most of those that do mention money refer to the contributors’ eligibility for a share of a generally available

— I
— o
— +I

prize for solving a scientific or mathematical challenge, and mix an appeal to hobby and altruism with the promise of money. Only two of about sixty projects active in 2004 were built on a pay-per-contribution basis, and these were quite small-scale by comparison to many of the others.

Most of the distributed computing projects provide a series of utilities and statistics intended to allow contributors to attach meaning to their contributions in a variety of ways. The projects appear to be eclectic in their implicit social and psychological theories of the motivations for participation in the projects. Sites describe the scientific purpose of the models and the specific scientific output, including posting articles that have used the calculations. In these components, the project organizers seem to assume some degree of taste for generalized altruism and the pursuit of meaning in contributing to a common goal. They also implement a variety of mechanisms to reinforce the sense of purpose, such as providing aggregate statistics about the total computations performed by the project as a whole. However, the sites also seem to assume a healthy dose of what is known in the anthropology of gift literature as agonistic giving—that is, giving intended to show that the person giving is greater than or more important than others, who gave less. For example, most of the sites allow individuals to track their own contributions, and provide “user of the month”-type rankings. An interesting characteristic of quite a few of these is the ability to create “teams” of users, who in turn compete on who has provided more cycles or work units. SETI@home in particular taps into ready-made nationalisms, by offering country-level statistics. Some of the team names on Folding@home also suggest other, out-of-project bonding measures, such as national or ethnic bonds (for example, Overclockers Australia or Alliance Francophone), technical minority status (for example, Linux or MacAddict4Life), and organizational affiliation (University of Tennessee or University of Alabama), as well as shared cultural reference points (Knights who say Ni!). In addition, the sites offer platforms for simple connectedness and mutual companionship, by offering user fora to discuss the science and the social participation involved. It is possible that these sites are shooting in the dark, as far as motivating sharing is concerned. It also possible, however, that they have tapped into a valuable insight, which is that people behave sociably and generously for all sorts of different reasons, and that at least in this domain, adding reasons to participate—some agonistic, some altruistic, some reciprocity-seeking—does not have a crowding-out effect.

Like distributed computing projects, peer-to-peer file-sharing networks are

— I
— O
— +I

an excellent example of a highly efficient system for storing and accessing data in a computer network. These networks of sharing are much less “mysterious,” in terms of understanding the human motivation behind participation. Nevertheless, they provide important lessons about the extent to which large-scale collaboration among strangers or loosely affiliated users can provide effective communications platforms. For fairly obvious reasons, we usually think of peer-to-peer networks, beginning with Napster, as a “problem.” This is because they were initially overwhelmingly used to perform an act that, by the analysis of almost any legal scholar, was copyright infringement. To a significant extent, they are still used in this form. There were, and continue to be, many arguments about whether the acts of the firms that provided peer-to-peer software were responsible for the violations. However, there has been little argument that anyone who allows thousands of other users to make copies of his or her music files is violating copyright—hence the public interpretation of the creation of peer-to-peer networks as primarily a problem. From the narrow perspective of the law of copyright or of the business model of the recording industry and Hollywood, this may be an appropriate focus. From the perspective of diagnosing what is happening to our social and economic structure, the fact that the files traded on these networks were mostly music in the first few years of this technology’s implementation is little more than a distraction. Let me explain why.

Imagine for a moment that someone—be it a legislator defining a policy goal or a businessperson defining a desired service—had stood up in mid-1999 and set the following requirements: “We would like to develop a new music and movie distribution system. We would like it to store all the music and movies ever digitized. We would like it to be available from anywhere in the world. We would like it to be able to serve tens of millions of users at any given moment.” Any person at the time would have predicted that building such a system would cost tens if not hundreds of millions of dollars; that running it would require large standing engineering staffs; that managing it so that users could find what they wanted and not drown in the sea of content would require some substantial number of “curators”—DJs and movie buffs—and that it would take at least five to ten years to build. Instead, the system was built cheaply by a wide range of actors, starting with Shawn Fanning’s idea and implementation of Napster. Once the idea was out, others perfected the idea further, eliminating the need for even the one centralized feature that Napster included—a list of who had what files on which computer that provided the matchmaking function in the Napster

— I
— o
— + I

network. Since then, under the pressure of suits from the recording industry and a steady and persistent demand for peer-to-peer music software, rapid successive generations of Gnutella, and then the FastTrack clients KaZaa and Morpheus, Overnet and eDonkey, the improvements of BitTorrent, and many others have enhanced the reliability, coverage, and speed of the peer-to-peer music distribution system—all under constant threat of litigation, fines, police searches and even, in some countries, imprisonment of the developers or users of these networks.

What is truly unique about peer-to-peer networks as a signal of what is to come is the fact that with ridiculously low financial investment, a few teenagers and twenty-something-year-olds were able to write software and protocols that allowed tens of millions of computer users around the world to cooperate in producing the most efficient and robust file storage and retrieval system in the world. No major investment was necessary in creating a server farm to store and make available the vast quantities of data represented by the media files. The users' computers are themselves the "server farm." No massive investment in dedicated distribution channels made of high-quality fiber optics was necessary. The standard Internet connections of users, with some very intelligent file transfer protocols, sufficed. Architecture oriented toward enabling users to cooperate with each other in storage, search, retrieval, and delivery of files was all that was necessary to build a content distribution network that dwarfed anything that existed before.

Again, there is nothing mysterious about why users participate in peer-to-peer networks. They want music; they can get it from these networks for free; so they participate. The broader point to take from looking at peer-to-peer file-sharing networks, however, is the sheer effectiveness of large-scale collaboration among individuals once they possess, under their individual control, the physical capital necessary to make their cooperation effective. These systems are not "subsidized," in the sense that they do not pay the full marginal cost of their service. Remember, music, like all information, is a nonrival public good whose marginal cost, once produced, is zero. Moreover, digital files are not "taken" from one place in order to be played in the other. They are replicated wherever they are wanted, and thereby made more ubiquitous, not scarce. The only actual social cost involved at the time of the transmission is the storage capacity, communications capacity, and processing capacity necessary to store, catalog, search, retrieve, and transfer the information necessary to replicate the files from where copies reside to where more copies are desired. As with any nonrival good, if Jane is willing

— I
— O
— +I

to spend the actual social costs involved in replicating the music file that already exists and that Jack possesses, then it is efficient that she do so without paying the creator a dime. It may throw a monkey wrench into the particular way in which our society has chosen to pay musicians and recording executives. This, as we saw in chapter 2, trades off efficiency for longer-term incentive effects for the recording industry. However, it is efficient within the normal meaning of the term in economics in a way that it would not have been had Jane and Jack used subsidized computers or network connections.

As with distributed computing, peer-to-peer file-sharing systems build on the fact that individual users own vast quantities of excess capacity embedded in their personal computers. As with distributed computing, peer-to-peer networks developed architectures that allowed users to share this excess capacity with each other. By cooperating in these sharing practices, users construct together systems with capabilities far exceeding those that they could have developed by themselves, as well as the capabilities that even the best-financed corporations could provide using techniques that rely on components they fully owned. The network components owned by any single music delivery service cannot match the collective storage and retrieval capabilities of the universe of users' hard drives and network connections. Similarly, the processors arrayed in the supercomputers find it difficult to compete with the vast computation resource available on the millions of personal computers connected to the Internet, and the proprietary software development firms find themselves competing, and in some areas losing to, the vast pool of programming talent connected to the Internet in the form of participants in free and open source software development projects.

In addition to computation and storage, the last major element of computer communications networks is connectivity. Here, too, perhaps more dramatically than in either of the two other functionalities, we have seen the development of sharing-based techniques. The most direct transfer of the design characteristics of peer-to-peer networks to communications has been the successful development of Skype—an Internet telephony utility that allows the owners of computers to have voice conversations with each other over the Internet for free, and to dial into the public telephone network for a fee. As of this writing, Skype is already used by more than two million users at any given moment in time. They use a FastTrack-like architecture to share their computing and communications resources to create a global

— I
— O
— + I

telephone system running on top of the Internet. It was created, and is run by, the developers of KaZaa.

Most dramatically, however, we have seen these techniques emerging in wireless communications. Throughout almost the entire twentieth century, radio communications used a single engineering approach to allow multiple messages to be sent wirelessly in a single geographic area. This approach was to transmit each of the different simultaneous messages by generating separate electromagnetic waves for each, which differed from each other by the frequency of oscillation, or wavelength. The receiver could then separate out the messages by ignoring all electromagnetic energy received at its antenna unless it oscillated at the frequency of the desired message. This engineering technique, adopted by Marconi in 1900, formed the basis of our notion of “spectrum”: the range of frequencies at which we know how to generate electromagnetic waves with sufficient control and predictability that we can encode and decode information with them, as well as the notion that there are “channels” of spectrum that are “used” by a communication. For more than half a century, radio communications regulation was thought necessary because spectrum was scarce, and unless regulated, everyone would transmit at all frequencies causing chaos and an inability to send messages. From 1959, when Ronald Coase first published his critique of this regulatory approach, until the early 1990s, when spectrum auctions began, the terms of the debate over “spectrum policy,” or wireless communications regulation, revolved around whether the exclusive right to transmit radio signals in a given geographic area should be granted as a regulatory license or a tradable property right. In the 1990s, with the introduction of auctions, we began to see the adoption of a primitive version of a property-based system through “spectrum auctions.” By the early 2000s, this system allowed the new “owners” of these exclusive rights to begin to shift what were initially purely mobile telephony systems to mobile data communications as well.

By this time, however, the century-old engineering assumptions that underlay the regulation-versus-property conceptualization of the possibilities open for the institutional framework of wireless communications had been rendered obsolete by new computation and network technologies.¹¹ The dramatic decline in computation cost and improvements in digital signal processing, network architecture, and antenna systems had fundamentally changed the design space of wireless communications systems. Instead of having one primary parameter with which to separate out messages—the

— I
— O
— + I

frequency of oscillation of the carrier wave—engineers could now use many different mechanisms to allow much smarter receivers to separate out the message they wanted to receive from all other sources of electromagnetic radiation in the geographic area they occupied. Radio transmitters could now transmit at the same frequency, simultaneously, without “interfering” with each other—that is, without confusing the receivers as to which radiation carried the required message and which did not. Just like automobiles that can share a commons-based medium—the road—and unlike railroad cars, which must use dedicated, owned, and managed railroad tracks—these new radios could share “the spectrum” as a commons. It was no longer necessary, or even efficient, to pass laws—be they in the form of regulations or of exclusive property-like rights—that carved up the usable spectrum into exclusively controlled slices. Instead, large numbers of transceivers, owned and operated by end users, could be deployed and use equipment-embedded protocols to coordinate their communications.

The reasons that owners would share the excess capacity of their new radios are relatively straightforward in this case. Users want to have wireless connectivity all the time, to be reachable and immediately available everywhere. However, they do not actually want to communicate every few microseconds. They will therefore be willing to purchase and keep turned on equipment that provides them with such connectivity. Manufacturers, in turn, will develop and adhere to standards that will improve capacity and connectivity. As a matter of engineering, what has been called “cooperation gain”—the improved quality of the system gained when the nodes cooperate—is the most promising source of capacity scaling for distributed wireless systems.¹² Cooperation gain is easy to understand from day-to-day interactions. When we sit in a lecture and miss a word or two, we might turn to a neighbor and ask, “Did you hear what she said?” In radio systems, this kind of cooperation among the antennae (just like the ears) of neighbors is called antenna diversity, and is the basis for the design of a number of systems to improve reception. We might stand in a loud crowd without being able to shout or walk over to the other end of the room, but ask a friend: “If you see so and so, tell him x ”; that friend then bumps into a friend of so and so and tells that person: “If you see so and so, tell him x ”; and so forth. When we do this, we are using what in radio engineering is called repeater networks. These kinds of cooperative systems can carry much higher loads without interference, sharing wide swaths of spectrum,

— I
— O
— +I

in ways that are more efficient than systems that rely on explicit market transactions based on property in the right to emit power in discrete frequencies. The design of such “ad hoc mesh networks”—that is, networks of radios that can configure themselves into cooperative networks as need arises, and help each other forward messages and decipher incoming messages over the din of radio emissions—are the most dynamic area in radio engineering today.

This technological shift gave rise to the fastest-growing sector in the wireless communications arena in the first few years of the twenty-first century—WiFi and similar unlicensed wireless devices. The economic success of the equipment market that utilizes the few primitive “spectrum commons” available in the United States—originally intended for low-power devices like garage openers and the spurious emissions of microwave ovens—led toward at first slow, and more recently quite dramatic, change in U.S. wireless policy. In the past two years alone, what have been called “commons-based” approaches to wireless communications policy have come to be seen as a legitimate, indeed a central, component of the Federal Communication Commission’s (FCC’s) wireless policy.¹³ We are beginning to see in this space the most prominent example of a system that was entirely oriented toward regulation aimed at improving the institutional conditions of market-based production of wireless transport capacity sold as a finished good (connectivity minutes), shifting toward enabling the emergence of a market in shareable goods (smart radios) designed to provision transport on a sharing model.

I hope these detailed examples provide a common set of mental pictures of what peer production looks like. In the next chapter I explain the economics of peer production of information and the sharing of material resources for computation, communications, and storage in particular, and of nonmarket, social production more generally: why it is efficient, how we can explain the motivations that lead people to participate in these great enterprises of nonmarket cooperation, and why we see so much more of it online than we do off-line. The moral and political discussion throughout the remainder of the book does not, however, depend on your accepting the particular analysis I offer in chapter 4 to “domesticate” these phenomena within more or less standard economics. At this point, it is important that the stories have provided a texture for, and established the plausibility of,

— I
— O
— + I

90 The Networked Information Economy

the claim that nonmarket production in general and peer production in particular are phenomena of much wider application than free software, and exist in important ways throughout the networked information economy. For purposes of understanding the political implications that occupy most of this book, that is all that is necessary.

— I
— O
— +I

rather than the expectation of legal rights of exclusion. The absorption effect is described in Wesley M. Cohen and Daniel A. Leventhal, “Innovation and Learning: The Two Faces of R&D,” *The Economic Journal* 99 (1989): 569–596. The collaboration effect was initially described in Richard R. Nelson, “The Simple Economics of Basic Scientific Research,” *Journal of Political Economy* 67 (June 1959): 297–306. The most extensive work over the past fifteen years, and the source of the term of learning networks, has been from Woody Powell on knowledge and learning networks. Identifying the role of markets made concentrated by the limited ability to use information, rather than through exclusive rights, was made in F. M. Scherer, “Nordhaus’s Theory of Optimal Patent Life: A Geometric Reinterpretation,” *American Economic Review* 62 (1972): 422–427.

10. Eric von Hippel, *Democratizing Innovation* (Cambridge, MA: MIT Press, 2005).
11. Eben Moglen, “Anarchism Triumphant: Free Software and the Death of Copyright,” *First Monday* (1999), http://www.firstmonday.dk/issues/issue4_8/moglen/.

CHAPTER 3. Peer Production and Sharing

1. For an excellent history of the free software movement and of open-source development, see Glyn Moody, *Rebel Code: Inside Linux and the Open Source Revolution* (New York: Perseus Publishing, 2001).
2. Elinor Ostrom, *Governing the Commons: The Evolution of Institutions for Collective Action* (Cambridge: Cambridge University Press, 1990).
3. Josh Lerner and Jean Tirole, “The Scope of Open Source Licensing” (Harvard NOM working paper no. 02–42, table 1, Cambridge, MA, 2002). The figure is computed out of the data reported in this paper for the number of free software development projects that Lerner and Tirole identify as having “restrictive” or “very restrictive” licenses.
4. Netcraft, April 2004 Web Server Survey, http://news.netcraft.com/archives/web_server_survey.html.
5. Clickworkers Results: Crater Marking Activity, July 3, 2001, <http://clickworkers.arc.nasa.gov/documents/crater-marking.pdf>.
6. B. Kanefsky, N. G. Barlow, and V. C. Gulick, *Can Distributed Volunteers Accomplish Massive Data Analysis Tasks?* <http://www.clickworkers.arc.nasa.gov/documents/abstract.pdf>.
7. J. Giles, “Special Report: Internet Encyclopedias Go Head to Head,” *Nature*, December 14, 2005, available at <http://www.nature.com/news/2005/051212/full/438900a.html>.
8. <http://www.techcentralstation.com/111504A.html>.
9. Yochai Benkler, “Coase’s Penguin, or Linux and the Nature of the Firm,” *Yale Law Journal* 112 (2001): 369.
10. IBM Collaborative User Experience Research Group, History Flows: Results (2003), <http://www.research.ibm.com/history/results.htm>.
11. For the full argument, see Yochai Benkler, “Some Economics of Wireless Communications,” *Harvard Journal of Law and Technology* 16 (2002): 25; and Yochai Benkler, “Overcoming Agoraphobia: Building the Commons of the Digitally Networked Environment,” *Harvard Journal of Law and Technology* 11 (1998): 287. For an excellent

— I
— O
— + I

overview of the intellectual history of this debate and a contribution to the institutional design necessary to make space for this change, see Kevin Werbach, “Supercommons: Towards a Unified Theory of Wireless Communication,” *Texas Law Review* 82 (2004): 863. The policy implications of computationally intensive radios using wide bands were first raised by George Gilder in “The New Rule of the Wireless,” *Forbes ASAP*, March 29, 1993, and Paul Baran, “Visions of the 21st Century Communications: Is the Shortage of Radio Spectrum for Broadband Networks of the Future a Self Made Problem?” (keynote talk transcript, 8th Annual Conference on Next Generation Networks, Washington, DC, November 9, 1994). Both statements focused on the potential abundance of spectrum, and how it renders “spectrum management” obsolete. Eli Noam was the first to point out that, even if one did not buy the idea that computationally intensive radios eliminated scarcity, they still rendered spectrum property rights obsolete, and enabled instead a fluid, dynamic, real-time market in spectrum clearance rights. See Eli Noam, “Taking the Next Step Beyond Spectrum Auctions: Open Spectrum Access,” *Institute of Electrical and Electronics Engineers Communications Magazine* 33, no. 12 (1995): 66–73; later elaborated in Eli Noam, “Spectrum Auction: Yesterday’s Heresy, Today’s Orthodoxy, Tomorrow’s Anachronism. Taking the Next Step to Open Spectrum Access,” *Journal of Law and Economics* 41 (1998): 765, 778–780. The argument that equipment markets based on a spectrum commons, or free access to frequencies, could replace the role planned for markets in spectrum property rights with computationally intensive equipment and sophisticated network sharing protocols, and would likely be more efficient even assuming that scarcity persists, was made in Benkler, “Overcoming Agoraphobia.” Lawrence Lessig, *Code and Other Laws of Cyberspace* (New York: Basic Books, 1999) and Lawrence Lessig, *The Future of Ideas: The Fate of the Commons in a Connected World* (New York: Random House, 2001) developed a rationale based on the innovation dynamic in support of the economic value of open wireless networks. David Reed, “Comments for FCC Spectrum Task Force on Spectrum Policy,” filed with the Federal Communications Commission July 10, 2002, crystallized the technical underpinnings and limitations of the idea that spectrum can be regarded as property.

11. See Benkler, “Some Economics,” 44–47. The term “cooperation gain” was developed by Reed to describe a somewhat broader concept than “diversity gain” is in multiuser information theory.
12. *Spectrum Policy Task Force Report to the Commission* (Federal Communications Commission, Washington, DC, 2002); Michael K. Powell, “Broadband Migration III: New Directions in Wireless Policy” (Remarks at the Silicon Flatiron Telecommunications Program, University of Colorado at Boulder, October 30, 2002).

CHAPTER 4. The Economics of Social Production

1. Richard M. Titmuss, *The Gift Relationship: From Human Blood to Social Policy* (New York: Vintage Books, 1971), 94.
2. Kenneth J. Arrow, “Gifts and Exchanges,” *Philosophy & Public Affairs* 1 (1972): 343.

— I
— O
— +I