First Monday

Code, Culture and Cash: The Fading Altruism of Open Source Development

by David Lancashire

Abstract

The nexus of open source development appears to have shifted to Europe over the last ten years. This paper explains why this trend undermines cultural arguments about "hacker ethics" and "post-scarcity" gift economies. It suggests that classical economic theory offers a more succinct explanation for the peculiar international distribution of open source development: hacking rises and falls inversely to its opportunity cost. This finding throws doubt on the Schumpeterian assumption that the efficiency of industrial systems can be measured without reference to the social institutions that bind them.

Contents

Introduction

The open source movement poses a serious challenge to traditional political economy. For a field accustomed to explaining human behavior through game-theoretic models of resource-distribution, the willingness of individuals to produce code for little or no monetary benefit is a glaring Achilles Heel. And so not surprisingly, some advocates of open source have gone so far as to paint the movement as a fundamentally new mode of industrial organization, a post-materialist one in which hackers code software simply for the pleasure of doing so. The empirical evidence speaks for itself, these critics assert, and in ways not at all favorable to classical economic theory. Considering that the open source hacker is in many ways a peculiar beast, not only in his apparent disregard for the laws of production, but also in cultural matters generally, the silence of political scientists and economists on this question is perhaps understandable. With very few exceptions, no serious attempts have been made to reconcile classical economic theory with an utterly puzzling observation: popular support for free-software projects. And as a result, advocates of cultural arguments have achieved the intellectual equivalent of a *coup d'etat:* their assumptions about underlying developer motivations have become conventional wisdom. Where the world works, hackers play.

Given that significant decisions continue to be made on this understanding of the open source movement, it is discomforting to reflect on how little empirical research actually supports this point [1]. Simply put, although it is common to assume that cultural factors drive hackers to collaborate on open source projects, there is little trustworthy evidence to support such a claim. Part of this problem stems from the methodological crisis facing scholars of open source. The difficulty of gathering accurate data about developers has forced much of our knowledge about them to be culled from in-depth ethnographic case studies [2]. While these provide excellent grist for discussions of organizational processes, as in Eric Raymond's discussion of Brooks' Law and Ko Kuwabara's treatment of open source development as a complex adaptive process, they provide almost no leverage on questions about underlying causation. This is a crucial weakness. Individuals tend to rationalize their actions to present themselves in the best light possible. Developers may be altruistic, but accepting their claims of selflessness at face value not only trivializes this tendency, but also assumes that developers consciously understand the combination of forces impelling them to behave as they do. Tackling the debate over the true cause of open source development therefore demands a much more introspective approach than is found in any work to date. It requires an analysis capable of testing cultural factors against those

economic pressures classical theories posit as operating below the level of human consciousness.

This paper is not intended as a critique of cultural studies in general. At least in the open source literature, these arguments provide intelligent explanations for the actual *dynamic* of development. Their problem is that they offer very little explanatory power when applied to the most fundamental question of all: why does open source development occur in the first place? And here, evidence suggests that conventional wisdom is profoundly misleading. As an analysis of international contributions to open source projects reveals, a very few countries produce a completely disproportionate amount of free software. At least for the subset of open source projects most widely assumed to be driven by altruism, the nexus of development has shifted towards Europe over the last ten years. This is a crucial observation - for contending hypotheses about why developers contribute to these projects lead deductively to diverging expectations of where open source work should be concentrated internationally. And the most compelling explanation for the current skew is an economic one. Differentials in international market conditions appear to play a decisive role in determining the relative vibrancy of open source communities worldwide.

The remainder of this paper presents this argument in depth. The first section reviews the theoretical differences between "economic" and "cultural" theories of open source production. It explains where these two approaches diverge, and reveals why the vast majority of "economic" analyses to date violate the most basic assumptions of classical economic theory [3]. Because it is useful to examine these arguments in their historical context, this section analyzes the open source literature chronologically, beginning with the work of Eric S. Raymond before turning to more recent scholarship. The second section sets up an analytic framework to test these contending hypotheses. The third and fourth present the actual research findings, and explain why they suggest that free software production is responsive to market pressures.

Cultural and Economic Theories of Open Source Development

The earliest writings on open source have largely come from "true believers" in the movement. The most prominent of these is hacker-scribe Eric Raymond, whose seminal 1998 study of the Linux community entitled "The Cathedral and the Bazaar" created the framework within which OSS continues to be discussed today. And perhaps ironically, considering that Raymond is a trained mathematician, the dominant casual mechanism in his work is cultural [4]. Highlighting the difference between open source "hackers" and traditional software developers, Raymond argues that:

> "The 'utility function' Linux hackers are maximizing is not classically economic, but is the intangible [product] of their own ego satisfaction and reputation among other hackers."

While this may be a *rational-choice* explanation (it seems to assume that individuals are goal-oriented actors who rank-order their preferences), it is far from an economic one [5]. By portraying developers as driven at least in part by "intangible" desires, Raymond undermines the most critical assumption of classical theory: that all actions have a quantifiable opportunity cost, and that individuals can consequently act in ways that maximize their material welfare [6]. In other words, although his argument is couched in the language of economics, Raymond implicitly suggests that open source development occurs outside of the market.

This is a non-trivial assumption. By focusing on cultural factors, Raymond is led deductively to conclude that open source projects are most successful when they nurture a culture prone to community-building - one where the price of admission is presumably some contribution to an evolving code base. At least one critic has confused this message for a Marxist one [7]. While it is not - Marx being an economic determinist - the confusion is perhaps understandable [8]. Written for consumption by the very community it celebrates, "The Cathedral and the Bazaar" is very much a political manifesto, packed-tight with what Raymond refers to as the "Unix gospel of small tools, rapid prototyping and evolutionary programming". Raymond's infectious enthusiasm makes his work engaging, but confounds efforts to impose a clear logic on his argument, not only because he tries to be simultaneously practical and theoretical, but because he jumps between institutional, structural and sociological arguments with breathtaking rapidity. Among other things, he never clarifies whether his view of human nature is functional or constructivist. Are open source programmers toolmakers, driven to adopt projects out of an innate need to create something concrete, or actors conditioned to participate because the community itself instills them with a sense of public service? And what exactly is meant by the term "community" anyway? If it never addresses these fundamental points, Raymond's argument is perhaps the more seductive because of its omissions, which leave the reader free interpret them according to his or her private inclinations [9].

To be fair to Raymond, criticizing his writing as theoretically opaque misses its significance. His attempt, after all, is to explain "what it is that makes the Linux community such a fountain of good software," not to provide a roadmap to the human soul. "The Cathedral and the Bazaar" is groundbreaking primarily for conceptualizing open source development as an organizational strategy allowing software development to buck the constraints of Brooks' mythical man month. At least in this first paper, Raymond's assumptions about cultural causation are also innocuous enough to be irrelevant to his main argument. Yet they do not remain innocuous for long, and have since conditioned an entire school of thought apt to uncritically assume that open source development is a cultural phenomenon. This bias is so pervasive that even critics of Raymond, such as Nikolai Bezroukov, tend to disagree merely about what cultural factors drive collaboration. Bezroukov suggests, for instance, that hackers support open source development more out of hostility to Microsoft (M$) than from any inherent desire to code for free.

If latent at first, Raymond's cultural assumptions grow more explicit in his later pieces. In "Homesteading the Noosphere", Raymond embraces anthropological theory to argue that developers operate in a "post-scarcity gift culture". Living in an environment of "abundance", they are driven not to maximize wealth so much as to give it away [10]. A much more explicitly libertarian piece than "The Cathedral and the Bazaar", "Homesteading the Noosphere" also draws the most revolutionary conclusions found in the literature, suggesting that:

> "Ultimately, the industrial-capitalist mode of software production was doomed to be out competed from the moment capitalism began to create enough of a wealth surplus for many programmers to live in a post-scarcity gift culture."

Here again, hackers are portrayed as unique for their cultural proclivity to reject the materialist trappings of modern life. Once generalized, this argument should be strikingly familiar. Political scientists will recognize in it the "post-materialist" thesis made most strongly by Ronald Inglehart in the book *Culture Shift in Advanced Industrial Society* (1990). Once individuals acquire a certain "wealth surplus", the story goes, they behave in ways economic theory simply cannot explain.

The tendency to portray hackers as members of a new cultural vanguard is an honored one in the annals of computer science. This tradition stretches back at least as far as 1984, when Steven Levy popularized the notion of the "hacker ethic" in his book *Hackers: Heroes of the Computer Revolution*. As the viral growth of perhaps apocryphal tales like "The Story of Mel" in early USENET discussions attests, the mythos of hackers as a culturally distinct class predates Levy's work by a healthy margin [11]. Wherever its origins however, the hacker ethic is clearly economically irrational, and can be narrowly defined as "the belief that... it is an ethical duty of hackers to share their expertise by writing open-source code" [12]. Drawing on Max Weber's arguments about Protestant and Catholic work ethics, contemporary arguments about hacker ethics reassert the primacy of culture as the key determinant of individual behavior within markets. As the recent publication of Pekka Himanen's *The Hacker Ethic* (2001) suggests, with its foreword by Linus Torvalds and postscript by Manuel Castells, this thesis is dominant enough not only to merit a book-length exposition, but also to attract essays of support from two of the most prominent gurus of the "Information Age".

But why the lack of skepticism? It is hardly surprising to encounter sociological theories of causation in a community so renowned for its cultural introspection. One of the conventions of cypherpunk fiction, as found in Robert Heinlein's *The Moon is a Harsh Mistress* (1966) or William Gibson's *Neuromancer* (1984), for example, is the romanticization of engineers as "elite" actors in a society where social boundaries are drawn primarily in terms of technical expertise. This same theme probably explains the recent popularity of Neal Stephenson's *Snow Crash* (1992) and *Cryptonomicon* (1999) among its similar audience. Both books' protagonists become epic heroes in part because their technical mastery allows them to manipulate mass consumer networks. As Sherry Turkle suggested in the mid-1980s, this affinity hackers hold for cultural assertions of their uniqueness is probably a manifestation of the basic human need to imbue meaning into those activities which define the individual's place in society.

> "Engineering students place great value on those things - books, movies, ideas - that connect their concerns with something larger. *Star Wars* was loved for the way it offered a bridge, even if superficial, between high technology and a romantic humanism. Robert Pirsig's *Zen and the Art of Motorcycle Maintenance* and Samuel Florman's *The Existential Pleasures of Engineering* ... achieved cult status because they describe how intense relationships with technical objects can lead to reflections on the philosophical concerns of the larger culture" [13].

Although a detailed discussion of this relationship is far beyond the scope of this paper, the key point is that cultural theories about hacker behavior have not emerged overnight. They developed as the implicit preference of a community predisposed to assert its cultural distinction. Considering his status as author of "The History of Hackerdom", it is safe to assume that Raymond was not only exposed to these ideas, but that they played a formative role shaping his view of

the world, and his arguments geared to explain open source.

And despite the frenzy of research that Raymond's work has sparked, very few scholars have challenged his cultural foundations. Many in the first wave of critics to embrace the open source movement appear to have done so *because* the phenomenon offered a strong empirical challenge to economic theory. Richard Barbrook's panegyric on the Internet as an "advanced form of social democracy", might more accurately be labeled an unrestrained broadside against neoliberalism, which Barbrook sees as only encouraging the "commodification of information" and the strangulation of "network communities ... created by gifts of time and ideas" [14]. Other early discussions of "gift cultures" are more apolitical, as in the work of Hillary Bays, but ignore the distinction between economic and cultural causation simply because their very starting point is cultural analysis.

Other early critics downplayed the chasm between cultural and economic theories in order to develop a broader understanding of the development cycle as a whole. In his attempt to extend Raymond's logic into a formal model, Rishab Ghosh stumbles only because the core logic of his argument requires basic development to have already occurred [15]. He cannot explain why individuals initiate projects other than to revert to Raymond's assumption that they do so to "scratch an itch". As Steve Weber elaborates in his review of this school, Ghosh fails to overcome the collective action problem first identified by Mancur Olson in his seminal *The Logic of Collective Action* (1971). As a result, although Ghosh provides an intelligent explanation for why existing projects gain snowballing support, he can neither predict *where* and *when* new development is likely to occur, nor explain *why* specific projects succeed where others fail.

The most reputable challenge to the cultural thesis is found in the work of economists Josh Lerner and Jean Tirole. In their NBER paper "The Simple Economics of Open Source", Lerner and Tirole draw lessons from the Apache and Perl cases to argue that individuals participate in open source projects only if they expect to derive "a net benefit" from doing so. Instead of seeing phenomena like "reputation games" as ends in themselves, Lerner and Tirole view them as means-to-an-end. They suggest that is it possible to quantify the "intangible" values Raymond and others ascribe to psychological needs such as "ego-boosting" - a necessary step in assigning open source hacking an opportunity cost and thereby salvaging economic theory. In their delicate *pas de deux,* Lerner and Tirole remind us that battles over reputation may in fact be zero-sum conflicts which "signal" the winner's technical expertise, and can be leveraged to gain privileged employment, access to venture capital, and the like.

A Framework for Analysis

In short, the analytic problem is that both cultural and economic explanations seem perfectly intuitive. When Alan Cox or Linus Torvalds defend the joy of programming, it seems absurd to suggest that their enthusiasm is feigned. The palpable sense of community on Linux-related Web sites like Slashdot and Kuro5hin further hints at the cultural solidarity of open source developers. And yet there is no denying that the very communities so quick to celebrate the open source movement have in the past been those quickest to "cash-in" on the phenomenon. Slashdot is part of the Open Source Developers Network (OSDN), and it is hardly coincidental that the site cheerleads for sister company Sourceforge when the stock price of parent company VA Linux swings with the productivity of unpaid developers. If cultural arguments hold merit, it seems equally plausible to argue that the true causal arrow points the other way, and that dynamism in the open source community is *the product* of wealth generation in the American software industry. Hackers may flock to open source largely because of the wealth potential of programming. Corporate interests might influence developer behavior, and culture may be better conceptualized as an intermediate (and not true casual) variable. How to untangle this Gordian knot?

And herein lies the problem. For obvious reasons, it is impossible to make robust arguments about causation when causal factors are confounded. From an analytic perspective, this confusion strongly suggests that the key problem the field faces is a lack of clarity over what constitutes the phenomenon under investigation. The logical solution is therefore to reduce the scope of our inquiry to projects most representative of the *kind* of open source development we seek to explain. Restricting our focus is doubly-necessary since over-generalizing about open source software can only lead to imprecise claims about the significance of the movement. We should hardly believe all projects share the same casual mechanisms, or are equally significant over the long-term. GNU/Linux may challenge Microsoft's monopoly over the consumer desktop, but few other projects are likely to have as telling an effect on commercial software providers, and thereby represent truly alternate forms of industrial organization. In terms of complexity and

accessibility, open source projects run the gamut from near-trivial cgi-scripts of less than one hundred lines of code to extremely-complex operating systems such as Aethos and Linux, the latter of which is currently over three million lines long [16]. Other factors likely to influence development include the operating context of and technical sophistication required to contribute to projects. While many programmers may code software out their desire to "scratch an itch", as even Lerner and Tirole concede in their analysis of their cases, some projects are driven by the involvement of commercial interests. Any history of GNU/Linux which ignores the vital role commercial distributors such as Red Hat and Mandrake-Linux played in standardizing and simplifying distributions for the home market would miss a large piece of the puzzle. Should growth in software packages funded by commercial firms be attributed to unpaid volunteerism? And confounding the situation yet further, as Steve Weber points out in his analysis, is that even trivial differences in the general license under which code is released affect subsequent development decisions in unrelated projects, creating path-dependent trajectories of software growth [17]. With all of these factors bundled into the debate over how open source works, it is hardly surprising to find such uncertainty in the literature.

Avoiding these traps requires us to test our two theories using a crucial case approach [18]. Selecting case studies in an *ad hoc* fashion is counterproductive. Cases should only be selected which *best represent* the phenomenon under investigation. In this case - since we are evaluating the empirical validity of challenges to economic theory - the critical projects are those most universally cited as proof of non-economic rationality on the part of developers. To help identify these cases, the following matrix presents what I believe is a reasonable typology of open source projects [19]. The vertical axis in this matrix groups projects according to their relationship to the private market. I classify as "Anti-Proprietary" those projects for which a strong, relatively low-cost commercial alternative *already* existed at the time of initial development. "Complementary" projects are those which either lacked established commercial competitors, or gained early support from corporate partners. The horizontal axis segments projects according to their overall complexity, both in terms of *code-complexity* (which scales with the size of the project and total number of functions) and *organizational-complexity* (which scales with the total number of developers).

	ANTI-PROPRIETARY	**COMPLEMENTARY**
HIGH COMPLEXITY		
LOW COMPLEXITY		

The crucial quadrant is the upper left-hand one. Non-complex software poses little threat to commercial software providers, since the barriers to entry in these markets are already so low as to preclude effective rent-capture [20]. Complementary projects make equally poor case studies as they are difficult to objectively analyze. The presence of admitted commercial interests in their development confounds attempts to isolate the importance of social factors. This is the weakness of the Lerner and Tirole study. While they demonstrate that economic factors played a vital role in fostering the growth of Apache and Perl, considering the relative position of these projects in our typology their conclusions are not only somewhat intuitive, but conceptually difficult to extend to other projects. It is certainly counterintuitive to extend their analysis to software released by the Free Software Foundation, where support for open source projects is closely linked to an ideological belief that free software is a natural outgrowth of free speech.

The combination of highly-complex and anti-proprietary projects offers the only quadrant in which the tension between economic and cultural assumptions about underlying human behavior can meaningfully be compared. It is an unfortunate fact then, if a somewhat revealing one on its own, that there are so few successful projects which fall into this category. Linux, an operating system begun in 1991 in order to provide a free alternative to commercial UNIX systems, is the most prominent example. The second-most so is undoubtedly GNOME, a free graphical-user interface (GUI) for UNIX-compatible systems begun in 1996 to compete with the partly privately-owned K-Desktop Environment (KDE) suite for UNIX and the completely proprietary Microsoft Windows [21]. With a combined total of over 430 developers, no other two projects approach the "authority" of these cases as benchmark examples of their kind, and therefore *most-favorable* cases for sociocultural criticisms of economic theory. If strong arguments supporting economic causation can be made for these two projects, as later sections of this paper attempt to provide, we should treat theories about cultural causation with much greater skepticism.

The Raw Data

The data available on the Gnome and Linux projects is revealing. For the purpose of analysis, I collected information on the country of residence for key contributors to the two projects. In the case of Linux, I relied on information located in the CREDITS file of all major kernel releases (from version 1.0 to version 2.4.9) [22]. For Gnome, I gathered developer-contact information from the project's web-site. Where information on the home-country of developers was not explicitly available, I performed private research to ascertain said information, or - in the last-case scenario - trusted information in the provided e-mail address of developers to infer home-country from domain ownership [23]. In the case of Linux, to avoid bias that might be introduced over time as developers migrated internationally, developers are continually counted as residents of the countries they were associated with when their names first entered the CREDITS file. Linus Torvalds, for instance, is continually listed as a Finnish developer, despite his recent migration to the United States to work with Transmeta. To minimize human error throughout, all data-collection and spread-sheet data-entry was automated by means of customized Perl and bash/sh scripts, with the output subsequently hand-checked to ensure accuracy.

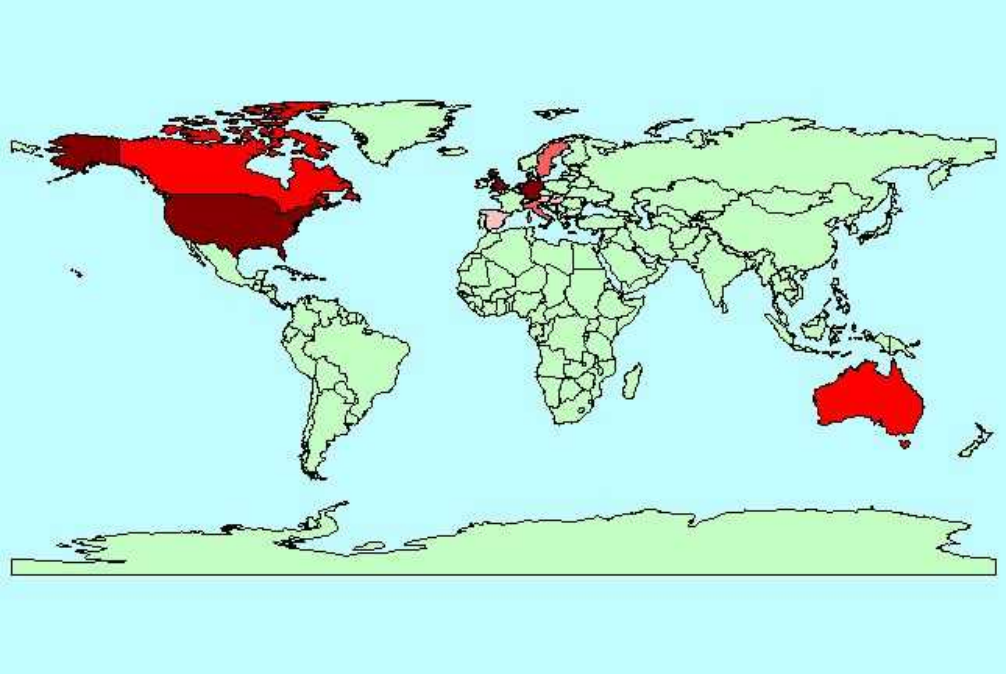


Figure 1: Absolute Linux Developers Worldwide
All maps are color-coded by quintile. Darker colored countries are more active in Open Source development. Calculations only include countries with greater than two developers in any project. Actual delimiting points vary map to map. See the Appendix for supporting data.

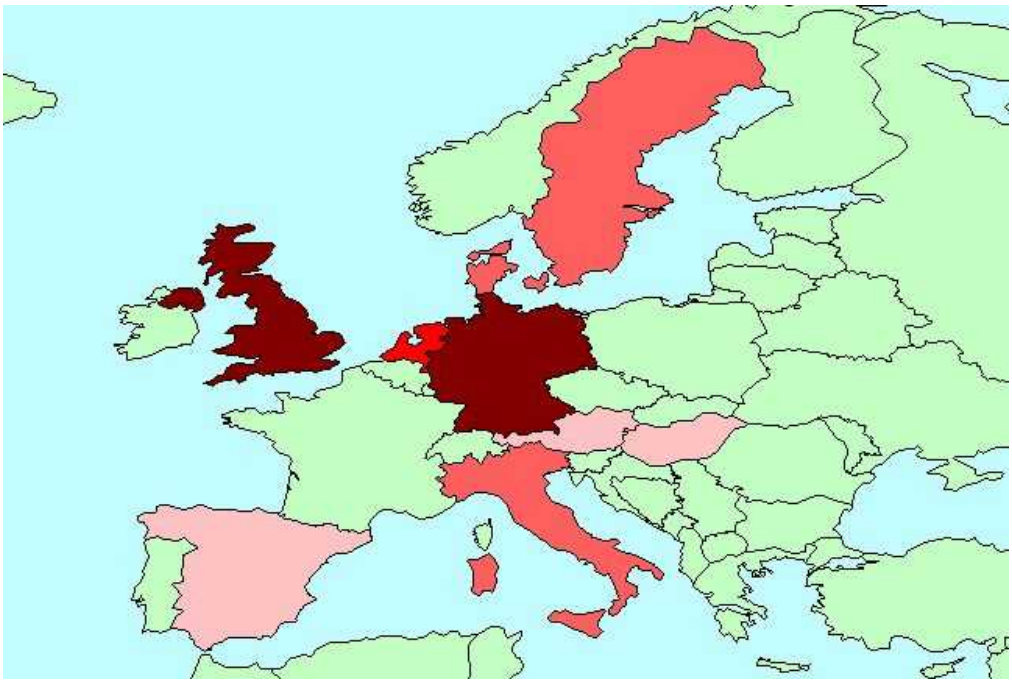


Figure 2: Absolute Linux Developers in Europe

Several immediate observations are striking. The first is the apparent dominance of the United States over free software development. The country contributes more than twice the number of developers to Linux as its closest runner-up, and nearly three-times as many to the Gnome project as any other country. A brief glance at the data confirms that large contributors in absolute terms also tend to be developed industrial countries with legacies of international openness and highly-educated, English-speaking populations. Northern Europe is particularly well represented. Not a single developer in either project comes from the Middle East, while all of Africa contributes but two in total - one from Egypt and one from South Africa. Asian and South American countries - presumably because of language barriers - are also underrepresented. Open Source development *does* appear - on first glance - to be highly correlated with "post-industrial" states and high levels of material development, an observation that supports the post-scarcity hypothesis.

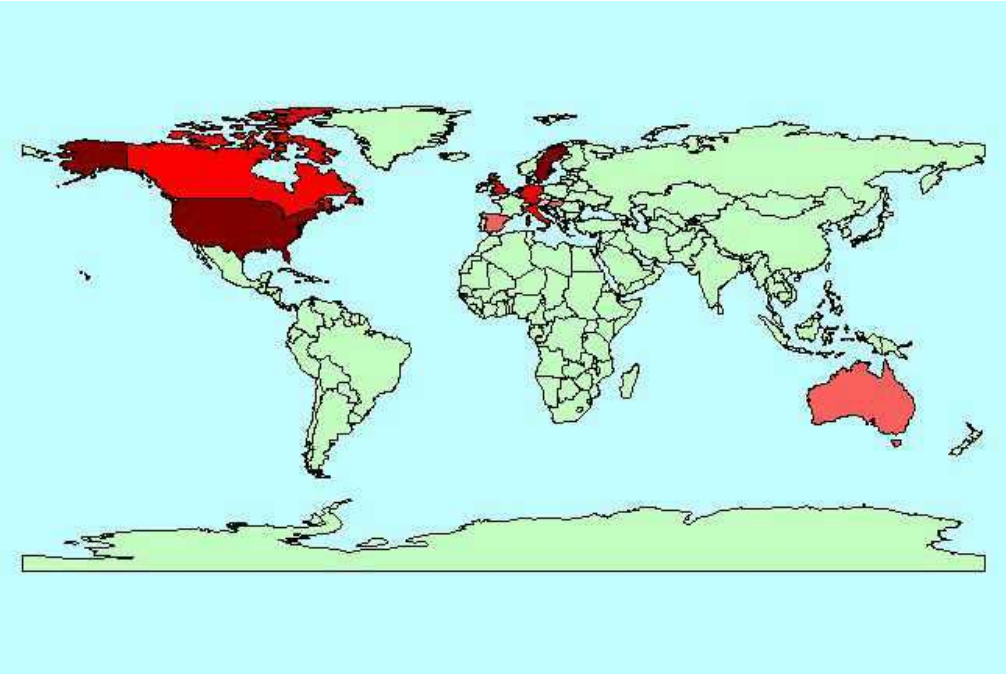


Figure 3: Absolute Gnome Developers Worldwide

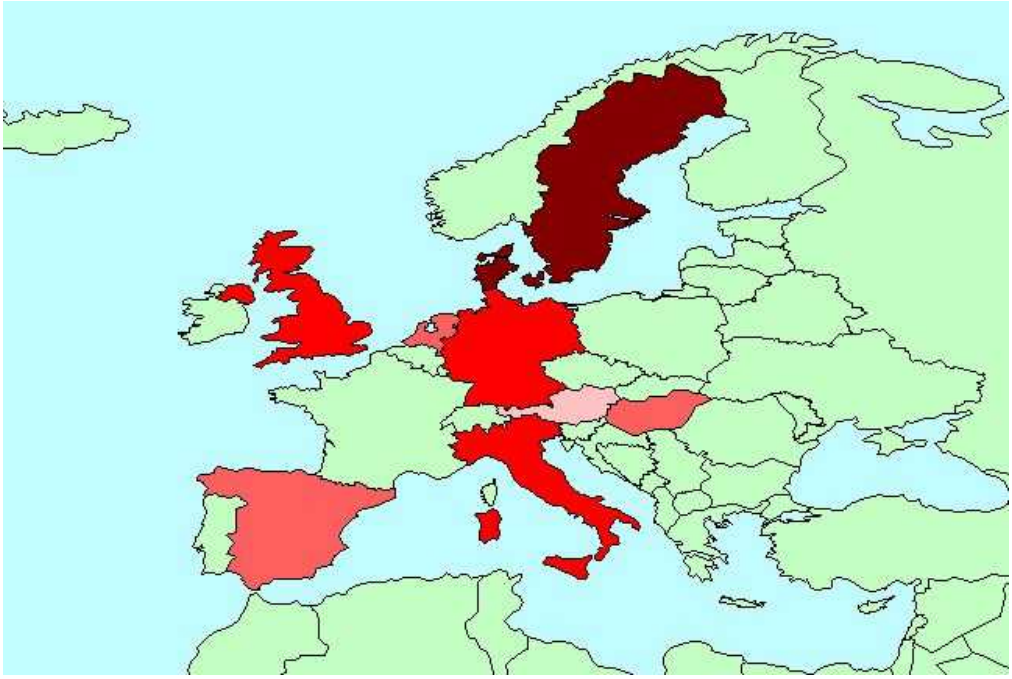


Figure 4: Absolute Gnome Developers in Europe

Less visible but also interesting is the fact that while most nations significantly active (developers > 1) in one project tend to be active in the other, smaller and less-developed nations cluster in support of particular projects. Brazil and the Czech Republic, for instance, contribute nine and 10 developers to the Linux project respectively, but not a single soul to Gnome. Mexico contributes three times as many developers to Gnome as Linux, and Finland (perhaps understandably considering its status as the homeland of Linus Torvalds) appears unwaveringly in the Linux camp.

This tendency offers tentative support for those who claim that goegraphic location matters; even in an age of global electronic communication, national or sub-national communities appear to be influential determinants of which projects national developers support.

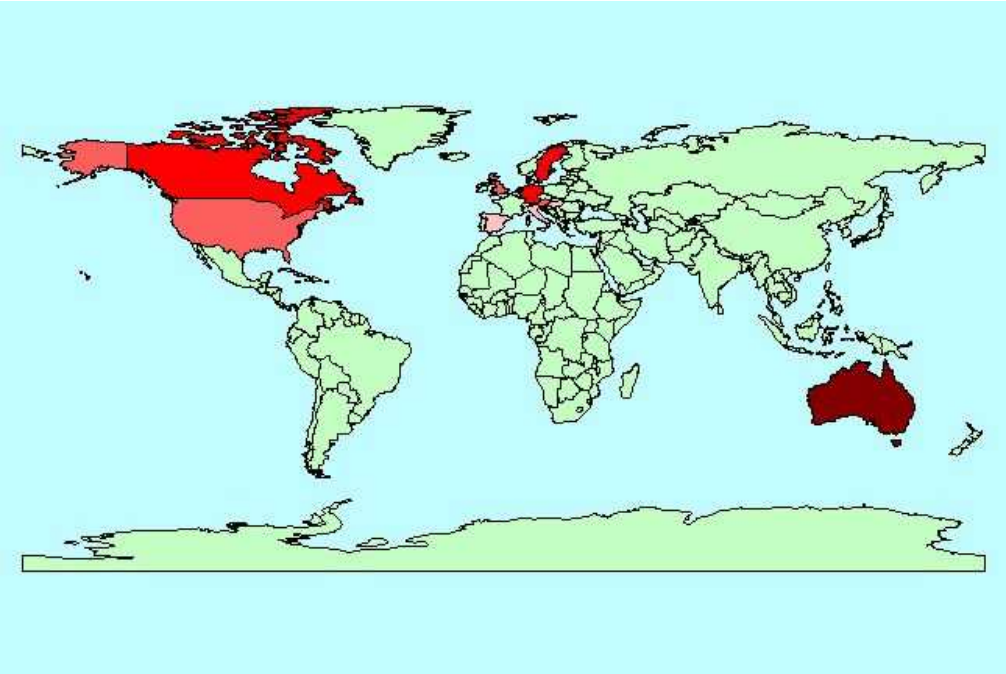


Figure 5: Linux Developers per capita Worldwide

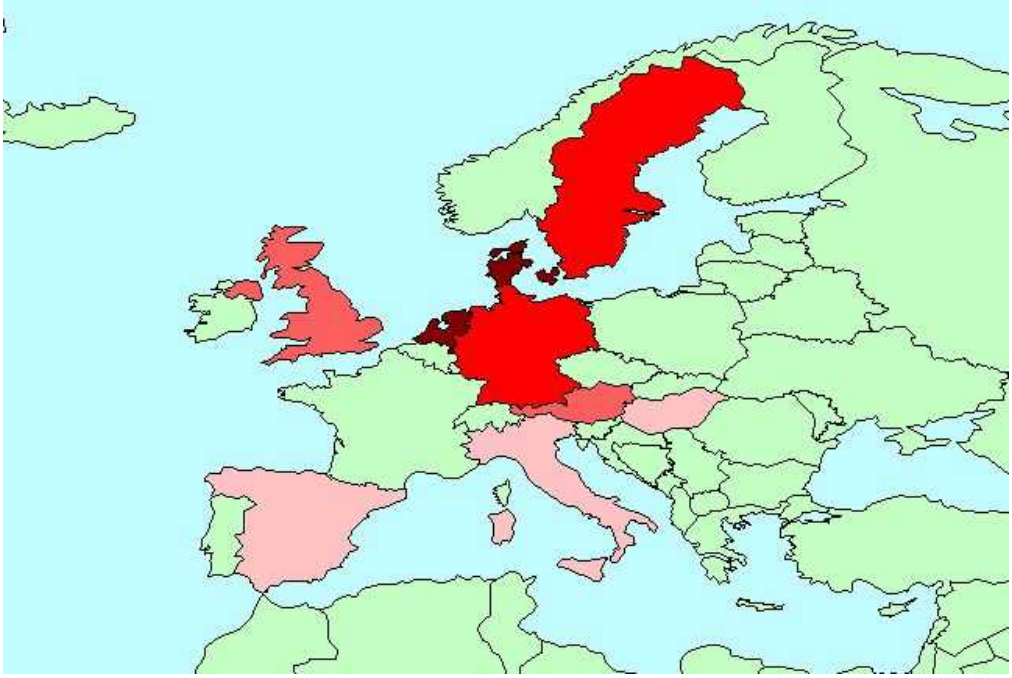


Figure 6: Linux Developers per capita in Europe

The problem with using absolute figures, however, is that they can be completely misleading. After all, the assumption of *ceteris paribus* in the economic model implies only that nations will contribute an equal number of developers to open source projects given their relative capabilities. And when our figures are adjusted to reflect population differences between nations, the list of "top contributors" to both projects changes significantly. When ranked with all other countries which contribute at least two developers to each project, the United States falls into an averaged tenth place overall - tied with Hungary and Spain [24]. Although it is possible to argue that the European-origins of the Linux project unfairly privilege Europe in these rankings, what little evidence we have suggests that the geographical origins of open source projects do not distort the distribution of developers in more populous nations. The United States contributes comparatively the same amount of effort to the European-based Linux initiative as it does to its own domestic Gnome project. Northern European countries are disproportionately represented among the top-tier of developers in both projects.

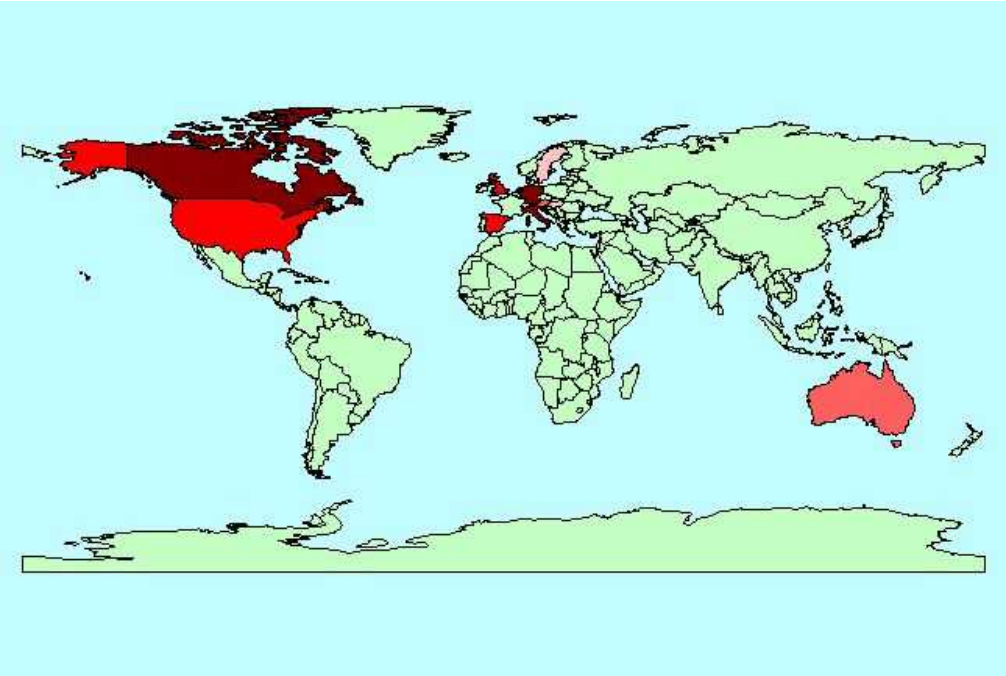


Figure 7: Gnome Developers per capita Worldwide

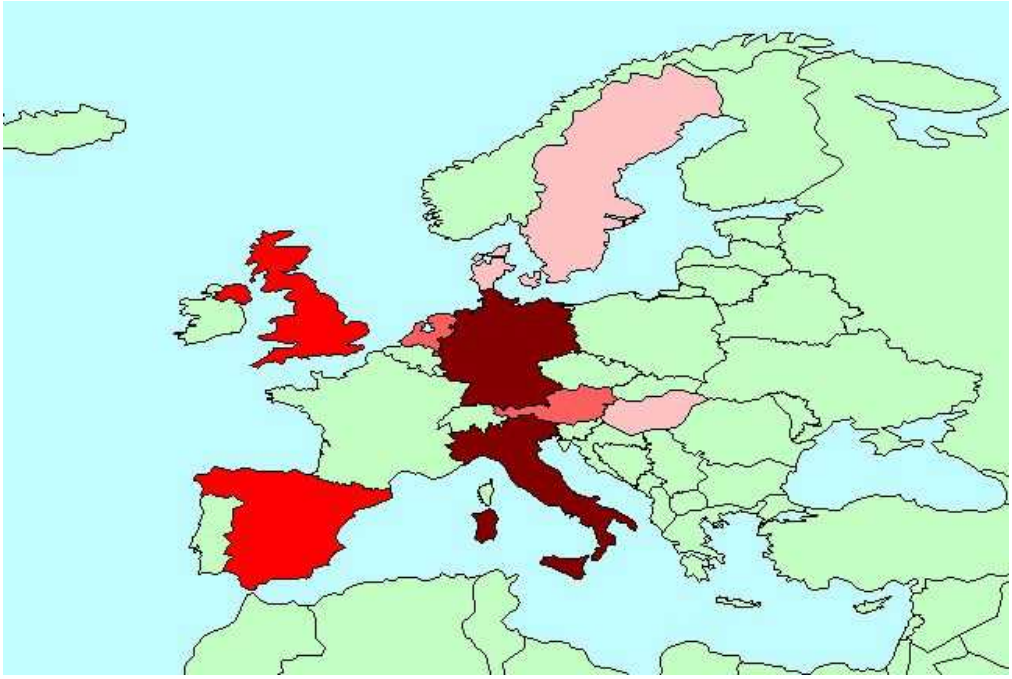


Figure 8: Gnome Developers per capita in Europe

And what makes these changes fascinating is that they are amplified when an even more precise proxy for the size of a country's computer-savvy population is substituted in place of simple population data. Obviously, the ability of a country to contribute to open source projects hinges not only on the size of its population, but on that of its sub-population with both the material resources and technical skills to engage in software development. Simple population data fails to differentiate between countries at different "stages" of development. It privileges small and extremely "wired" countries at the expense of larger and less-developed ones. Might this explain the disproportionate representation of the Northern European social democracies in our rankings? Is the United States simply less "developed" in terms of its technological sophistication?

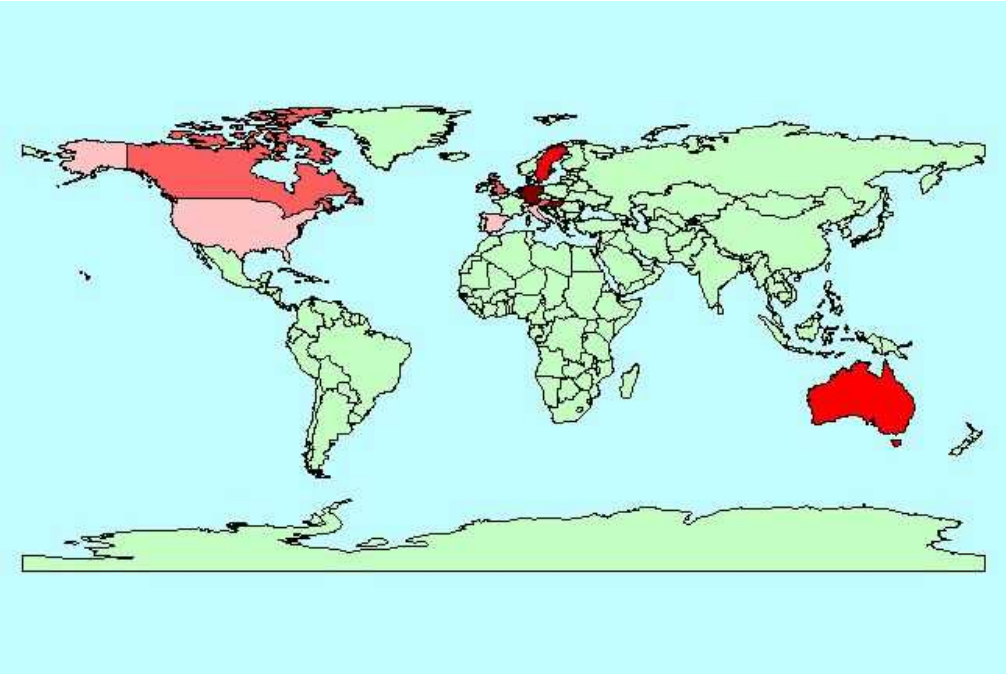


Figure 9: Linux Online per capita Worldwide

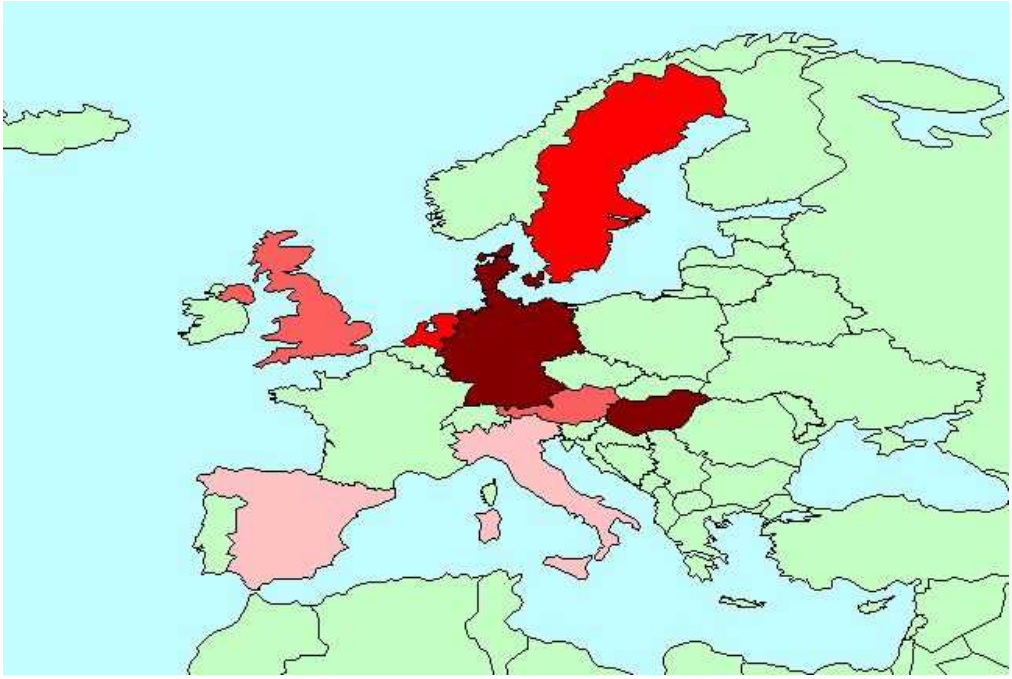


Figure 10: Linux Online per capita in Europe

To compensate for this potential bias, I recalculated these figures using estimates of the number of citizens with home access to the Internet in each country, an admittedly imprecise, but realistically much better estimate of the size of each nation's sub-population *capable* of undertaking actual development [25]. On the upside, this comparison should also remove some of the bias associated with international language barriers, considering that English is the *lingua franca* of the Internet. Admittedly, it risks privileging less-developed countries, where adopters of networking technologies are

more likely to hold advanced programming skills than those in more developed nations. After all, in extremely underdeveloped countries only the highly educated and extremely skilled are likely to be able to afford (or even want) access to global computing networks. Given that the quality of life in Europe and North America is roughly comparable, this bias should not distort our analysis.

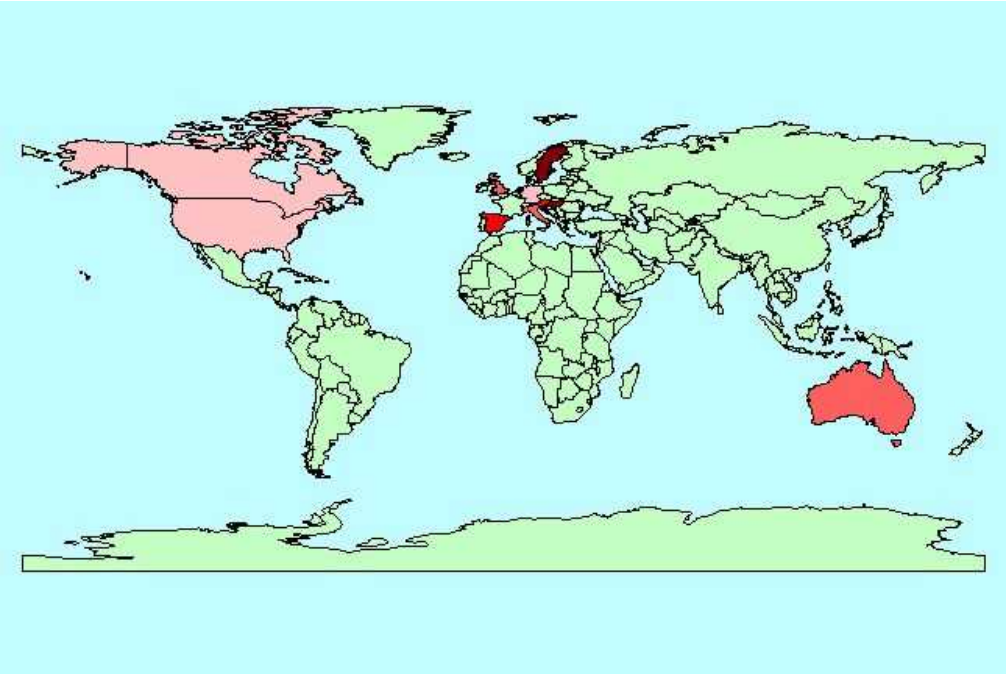


Figure 11: Gnome Online per capita Worldwide

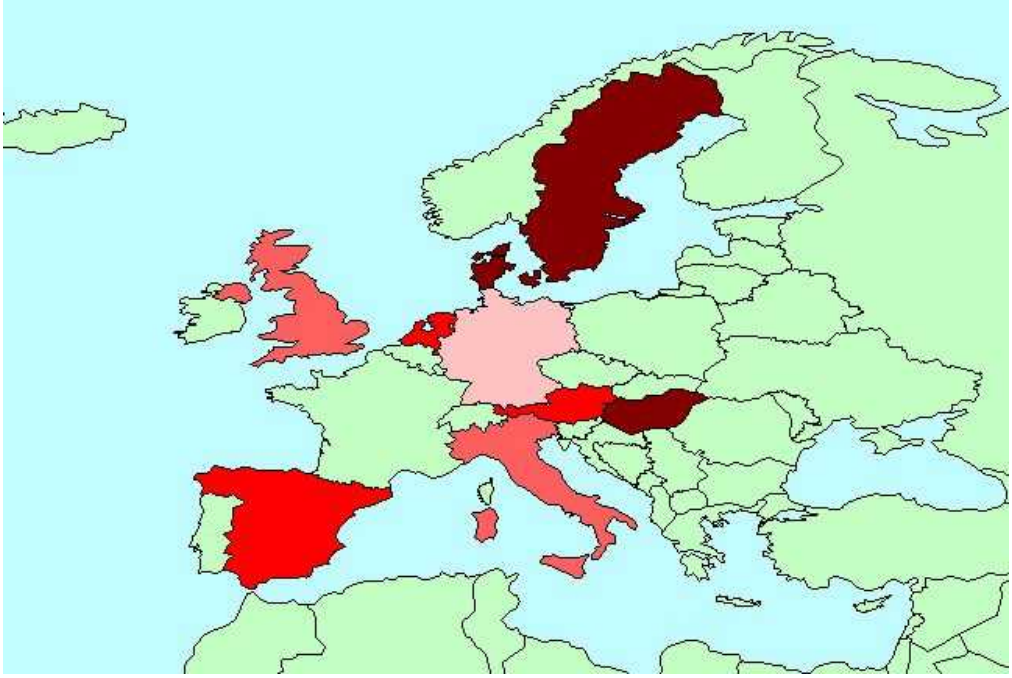


Figure 12: Gnome Online per capita in Europe

And our findings are remarkable. The United States consistently drops to a position of relatively inactive development - a very puzzling discovery. The Northern European social democracies hold their position at the head of the list, and appear to be undertaking a completely disproportionate amount of actual work. Countries in Southern Europe rank only slightly above the United States, while Mexico, Canada and the United Kingdom are positioned near the presumed mean. In traditional economic lingo, the United States appears to be free riding on a collective good provided predominantly by non-Americans. The stark nature of this contrast is a devastating blow to arguments about post-scarcity "gift cultures". If the wealthiest country in the world is one of the least active (relative) contributors to open source development, it seems ludicrous to explain said development as a function of post-materialism.

Economic Theory Revisited

With a relatively small sample size, this survey is not robust enough to draw statistically conclusive findings about *all* open source projects. But this is not the point. As has already been discussed, conducting a simple random survey of developers is completely infeasible since it invariably requires selecting on the dependent variable: the very criteria used to define what constitutes an open source project will inevitably introduce bias. It also seems likely that the international distribution of developers will change as we shift across the matrix presented above. Commercial support for complementary projects is undoubtedly stronger in the United States than in Europe, if only because American equity markets have historically supported open source companies. It is also probable that the availability of jobs in complementary projects has drawn American programmers away from *pro bono* efforts by offering them the best of both worlds: altruistic work coupled with a steady paycheck.

If it is difficult to generalize our conclusions to projects beyond the top-left quadrant however, this statistical analysis is more than robust enough ($n > 430$) to comment conclusively on the international distribution of Linux and Gnome developers. And if we accept the arguments of "gift culture" advocates that these developers are *most representative* of open source hackers as a whole, or at least that subset of hackers driven by "intangible" hacker ethics, the methodological problem inherent in selecting cases without a simple random sample diminishes considerably. In other words, by structuring this analysis in a way *most favorable* to advocates of cultural causation, this paper has been designed as structurally biased against the conclusions it nonetheless draws. And so to the extent that Linux and Gnome are representative of the kind of open source projects we seek to explain, we should accept these findings, which are so counterintuitive and consistent across our cases that they must spark efforts to explain them.

After all, conventional wisdom runs completely against this observed pattern. Historically, the United States has always been the country most commonly associated with the free software movement. From the late 1960s through the early 1980s, it was almost singlehandedly responsible for the vast majority of free software produced around the world. Early programs were given-away without-thought by academics and research scientists at leading American universities including MIT and UC Berkeley, and private research facilities such as Xerox PARC. The modularization of computer architectures around open standards in America further encouraged software-sharing in ways which never materialized abroad. Development in the rest of the world (and particularly in Europe and Japan) remained hidebound even into the early 1980s. At a time when Microsoft was gaining dominance over the American consumer market with early versions of MS-DOS, leading European firms such as Micro Focus continued to produce software for proprietary hardware architectures in "outdated" languages such as Cobol, and to target their goods to corporate - not consumer - markets [26]. Backing this record of technological leadership, the United States is also the undisputed center of an increasingly global culture prone to equate free software with political liberty. It is home to the world's most notable free software advocates, from Richard Stallman to the San Francisco Cypherpunks. As this record of cultural leadership is already well-documented in the work of Steve Weber and Stephen Levy, it suffices to ask a single question: how can we explain the apparent erosion of American support for free software development?

And here, cultural arguments provide little traction. It is absurd to argue that communities and cultures simply do not matter, and this paper does not make such a claim. Open source development to date has been dominated by English-speaking programmers communicating through English-based networks such as USENET and IRC, and in English dominated virtual communities such as Collab.net and SourceForge [27]. The stark non-participation of the non-English speaking world offers testament enough to this observation.

But there are significant problems with the cultural thesis. If individuals are drawn to support existing projects because

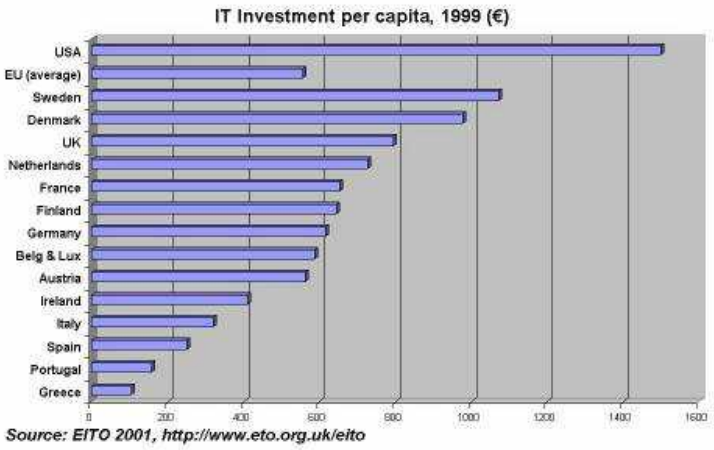
of cultural factors, communities which *begin* open source development should experience cascading support for free software initiatives, *not* a diminishment of support relative to countries with less "developed" communities. This is because of the "network effects" or endogenous growth hypothesis implicit in "gift culture" arguments. As open source communities expand, their ability to induce participation through the extension of intangible benefits like "reputation" and "strengthened identity" should expand as well. At the least, there is no easy way for universal arguments about hacker ethics to explain the strong variance in development across nations.

But for those willing to take political economy seriously, the shift of free software production to Europe suggests an underlying logic to developer behavior fully consistent with assumptions of pure economic rationality. Since this is a key point, I wish to make my argument here perfectly clear. Although often accused of it, political economy does not necessarily claim that individuals act directly on the basis of conscious cost-benefit calculations. Cultures and communities may be intermediating variables. Emotional factors such as the need for social acceptance and the desire for public recognition unquestionably influence human behavior. What theory suggests however, is that these social pressures may themselves be explained by underlying economic forces. Cultures may evolve in response to structural changes in the international economy. Slashdot may be popular exactly *because* the issues it covers are wrapped in the dynamism of the American high-tech economy, with its promise of attractive working conditions and high wages. Individuals may drift into hacker communities because they find the activities of those communities interesting, but the possibility exists that their interest itself might be conditioned by underlying economic forces which make certain types of activities economically rewarding and others less so. Highly prestigious trade associations rarely arise in minimum-wage industries.

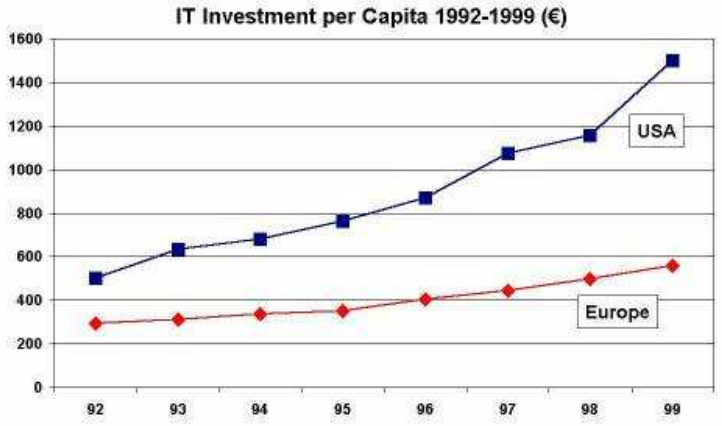
If this seems counterintuitive, it is useful to remember, as Milton Friedman once pointed out, that "to be important ... a hypothesis must be descriptively false in its assumptions" [28]. The claim is simple: any theory which challenges popular wisdom of how the world works is important *to the extent that it provides more accurate descriptive and predictive power* than those it seeks to replace. The test of any theory remains in its consistency with empirical observations about "real world" behavior, and its ability to offer a more thorough, parsimonious and compelling explanation for that behavior than its challengers. Any theory not *descriptively* false is unimportant because its lessons are self-evident.

Economic logic obviously provides a simplified model of human behavior. It is nonetheless capable of explaining the migration of open source development to Europe. Since the impetus for individuals to produce free software in these models is the expectation of tangible benefits sometime in the future, an economic analysis of open source development must pay attention to factors which influence the relative value of programmers' future earnings. On the demand-side, as long as barriers to international labor mobility exist the most important factor is clearly the relative vibrancy of national software industries [29]. As increased demand for programmers within a nation drives up the going wage rate, it should increase the opportunity cost of coding free software over commercial applications, and thereby decrease the amount of free software production [30].

And in this light, it is a near certainty that Americans face a higher opportunity cost for free software production than do Europeans. In the United States, the largest software exporting country in the world, the demand for highly-skilled computer professionals has soared in recent years, driving up wages and decreasing the attractiveness of doing for free that for which others are willing to pay extraordinary premiums. Considering the high barriers to entry in human-capital intensive industries, it seems likely that the proximate cause of this demand-hike has been the United States' extraordinary record of investment in its information technology infrastructure throughout the 1990s. The United States' level of IT investment has far exceeded its European counterparts' in this period. This difference is even more impressive if we accept that IT investments tend to have high multiplier effects in terms of their influence on labor market conditions. A single investment in an electronic database system creates high levels of demand *not only* for those producing the back-end hardware and software, but also for those with the skills to maintain the system and for third party providers of value-added services capable of extending it. One-time investments in enabling technologies create persistent patterns of demand for skilled technical workers.



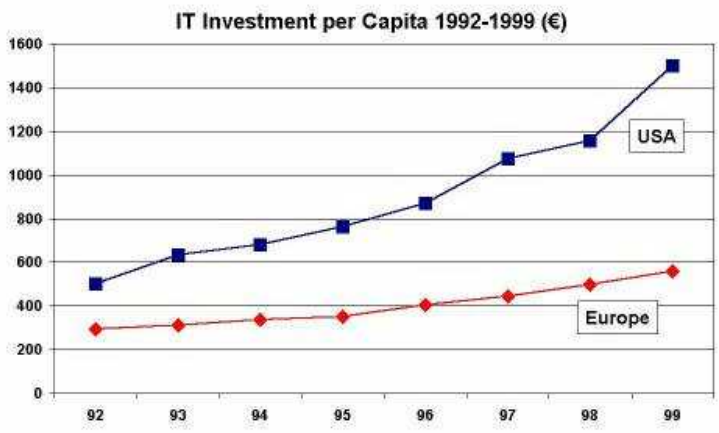
It is debatable what factors are responsible for the United States' ability to pour such staggering amounts of capital into its IT industry [31]. The key point strictly speaking is that the marginal utility of sector-specific labor in the United States has soared in the last ten years, driving up both productivity and (to the extent that there are barriers to entry in the IT labor market and that investment has a net positive effect on job creation) the average wage rage as well [32]. While the first assumption seems reasonable, the latter is unassailable. According to a 1999 survey of the American software industry conducted by Nathan Associates, core software development in the United States employed over 600,000 people in 1996, and has grown at a rate of almost 14 percent per year since then [33]. Although demand for computer programmers has grown at more modest nine percent, this high level of growth still indicates tremendous dynamism in the American software industry [34]. In contrast, Western European software industries remain relatively stagnant - and tend to focus on niche market segments and hardware dependent software services instead of the general pre-packaged software market with its fast-growing consumer base and extraordinary economies of scale.



Source: EITO 2001, http://www.eto.org.uk/eito

The relationship between the level of domestic investment and intensity of open source development can only be suggestive. A simple glance of the ordinal ranking of European nations in terms of their relative level of IT investment reveals occasional aberrations. Sweden, for instance, narrowly outperforms Germany in terms of its relative contribution to open source projects and yet maintains a slightly higher level of IT investment. And obviously, the lack of investment in countries such as Italy should hardly lead us to *expect* high levels of open source development any more than we should expect intense participation in the least developed parts of sub-Saharan Africa. Other factors are clearly at work. And yet, if attempting to run regression analysis on this question would be foolish, trying to force too

many variables into small a sample size, as Keynes once wrote on a completely different issue, "the general character of the facts presents itself irresistibly" [35].



Source: EITO 2001, http://www.eto.org.uk/eito

Institutional differences in the organizational structure of American and European software industries have exacerbated the two regions' economic divergence. As Trevor Nakagawa reminds us, only four of the top fifty software producers in 1998 were European, and 84 percent of these firms' revenues traditionally came from markets within the EU - markets heavily penetrated by American producers [36]. European firms lag significantly behind their American counterparts in terms of global software exports [37]. Social factors, such Europe's higher levels of software piracy, may be partially responsible for this difference. Whereas the Business Software Alliance estimates that 25 percent of all American business applications are pirated, its estimates of European piracy levels run from a low of 35 percent in Denmark and Germany, to over 50 percent in the Netherlands, and over 70 percent in Ireland and Greece [38]. Language barriers - which create smaller markets where amortizing development costs is more difficult - may also contribute to lower wages in Europe. National software development is simply less lucrative in smaller and less cutting-edge markets.

Table 1: Relative (%) Software Piracy Worldwide
Source: Business Software Alliance

Country	Piracy
United States	25.7
United Kingdom	30.7
Germany	32.3
Denmark	33.3
Austria	40.2
Australia	40.2
Canada	40.7
Sweden	42

Netherlands	48.8
Italy	49
Spain	59.8
Hungary	60

In short, the success of Silicon Valley in the 1990s cannot be understood without reference to broader economic pressures at the international level; American competitiveness in the pre-packaged software market flowed naturally from the United States' position as a world leader in modular computer hardware development [39]. The rise and fall of the American hacker cannot be understood in isolation either. Very significant improvements in the real income of high-tech workers followed the changes in worker "skill sets" demanded by the new forms of industrial organization first prevalent in the high-tech corridors of Boston and San Francisco. It seems likely that the future of the American high-tech industry and the popularity of open source hacking will depend on how patterns of labor use are encouraged by the consolidation of networking technologies or their extension in mass industrial production.

If this simplified model can explain the relative erosion of open source production in the United States, can it explain the rise of it Europe? Primarily, it should be clear that if the opportunity cost of working on open source projects is lower for European developers than their American counterparts, the potential benefits Europeans gain from working on them are much greater as well. In a global economy lacking perfect labor mobility and characterized by wage-inequality across countries, we expect individuals to produce free software *if* doing so can help them shift to a higher wage-level. This "fixed-cost" analysis implies (as Lerner and Tirole suggest in their paper) that developers may embrace open source work as a way to tap-into lucrative corporate networks abroad. This may explain why open source development is more popular in Canada than the United States, although the data from Europe is inconclusive on this question. This also helps to explain why the majority of open source developers are relatively young [40]. Older, settled programmers have less need to establish a monetizable reputation than their younger, more mobile counterparts, given less time in which to amortize its immediate costs.

In other words, the appropriate analogy for open source development may not be to "cooking-pots" and "cauldrons" so much as to the Mayflower. Addressing the similar debate over the causes of migration to the New World, Hatton and Williamson propose that:

> "Those who emigrated had the most to gain from the move, and were likely, therefore to be the most responsible to labor market conditions. By emigrating when young they were able to reap the gains over most of their working lives. By moving as single individuals they were able to minimize the costs of the move, including earnings forgone in passage and during job search" [41].

Conceptualizing open source work as a "fixed cost" young programmers face in order to establish a quantifiable reputation offers a convincing explanation for the peculiar demographics of open source development, and helps explain in particular why the phenomenon seems prevalent among young college-educated men. This nicely supports the observation made by Bezroukov that the romanticization of the open source movement in academic circles may have played a vital role in fuelling its early growth.

The analogy between open source development and historical migration is hardly perfect, but it is useful to consider even for its obvious differences. The most striking of these involves the type of labor assuming the fixed costs. Historically, the bulk of international migration has involved the extremely uneducated - those with few capital-invested skills, and little ability to earn a reasonable income in their local communities [42]. In contrast, contemporpary open source programmers can accurately be thought of as "hyper-skilled" and in many cases "over-qualified" for national markets. A programmer extremely knowledgeable about thread-management and memory protection under Unix, for instance, may find him or herself employed by the private sector in a completely different manner - writing Web-based applications or producing software through a mass-market integrated development environment (IDE). Establishing a reputation as a skilled specialist may help programmers transcend the limitations of their local markets - a pressure much more likely to influence decision-making in niche European markets with a less diverse production base.

Implications and Conclusions

If this hypothesis is supported by future research, a reinterpretation of the entire history of the free software movement will be necessary. For this analysis suggests a starkly different logic to open source development than is contained in most of the popular literature on the subject. In particular, it offers support for the hypothesis that early open source work thrived because its development took place in an immature and publicly-subsidized market. While academics and researchers were no doubt driven by a desire to "scratch an itch" and perform work they found particularly stimulating, it is significant that they essentially performed labor for which there was very little immediate private-market demand. Is free software truly free? It may be something for which developed countries have already paid: through early funding for academic research and development, and support for public research *at times when the market for certain types of software was immature*. It is hardly accidental that early "hacker" communities emerged at organizations with the resources and will to subsidize long-term development.

And although the role the United States government played in supporting early development in the American hardware-computing industry is generally recognized, its indirect contribution (and that of foreign governments) to the success of open source projects is usually completely overlooked in arguments which center on hacker ethics [43]. Part of this doubtless stems from the ideological biases of many commentators on this issue, whose political preferences have tended towards the libertarian. By portraying programmers as motivated primarily by cultural factors, early theorists could not only treat early developers as counterculture heroes in the cypherpunk tradition, but also paint the history of the open source movement as an all-American story of Jeffersonian independence. In one typical early interview, for instance, suggestions that open source projects might be explicable as "project[s] subsidized by government or academic" bodies were dismissed offhand by the interviewer. This rejection of classical economic logic is all the more remarkable given the overwhelming torrent of evidence that not only are many open source projects begun in academic circles, but that the growth of Linux itself was indirectly subsidized by the University of Helsinki [44]. It may be one of the odd ironies of computing history that the open source movement has proven such fertile grounds for libertarian philosophizing.

A more profound consequence of treating open source development from a classical perspective is the unexpected shadow it throws on one of the most cherished pillars of modern political economy. In particular, it opens the door to a devastating attack on the empirical validity of the principle of *creative destruction* invoked by Joseph Schumpeter in his opus *Capitalism, Socialism and Democracy* (1950). Here, Schumpeter does not (as is often assumed) simply assert that innovative businesses cannibalize their older counterparts. His argument is a more complex one about the process of innovation itself. Contrasting market and state-led production systems, Schumpeter argues that the process of innovation induced by market-systems is inevitably the more socially beneficial because capitalism can be *creative* in its ability to wring the last ounce of efficiency from industrial processes. Defending capitalism against the socialist encroachment he believed inevitable, Schumpeter pointed out that:

> "Since we are dealing with a process whose every element takes considerable time in revealing its true features and ultimate effects, there is no point in appraising the performance of that process *ex visu* of a given point in time; we must judge its performance over time, as it unfolds through decades or centuries."

This point was intended in defense of the short-term market inefficiencies Schumpeter recognized plagued American production in the 1940s. His defense rested on the belief that these failures in themselves created the long-term conditions necessary for efficient industrial development, by creating the incentives for further technical innovation. If one accepts the common view that open source software is more socially beneficial to society than commercial software, being more likely to lead to innovation because of its open and extensible nature, the success of projects like Linux in their twenty year competition with commercial alternatives creates a daunting empirical challenge to the Schumpeterian assumption that the efficiency of production systems can be measured without reference to the social systems that support them. If Raymond is right to conceptualize the *dynamic* of the Bazaar model as inherently more equitable than the Cathedral model, this criticism gains a moral edge as well. At least on first glance the historical record seems to suggest that - over decades - certain kinds of state intervention may *in fact* be socially desirable.

These questions are not merely academic: our understanding of how open source development actually works has profound implications for what kinds of corporate strategies and public policies firms and governments should pursue over time. If this paper is correct in its underlying conceptualization of developer motivations, projects such as Netscape's Mozilla which seek to leverage competitive advantage from unpaid public development are fool's errands - unlikely to create products which can be successfully leveraged for private profitability. Competition in service-

provision should also grow progressively unprofitable as expertise on open source products grows more publicly diffuse and Baumol's disease kicks-in. If open source production is stimulated by differentials in real income levels, corporate strategies that seek to ward-off open source competitors by hiring or co-opting their developers are equally prone to undermine themselves in the long-term, being fraught with moral hazard. By increasing the benefits which accrue to free software developers, strategies aimed at co-opting open source development will ironically serve only to *encourage* further development. And counterintuitively, this model predicts that periods of greatest prosperity in national software industries will correspond with increased activity among developers abroad - exactly the trend we seem to have observed. Will "hard times" in software industries reduce the attractiveness of coding free software abroad?

Unevenly-enforced social policies on issues such as software piracy and labor mobility become incredibly important areas of corporate concern and public interest in this light. By mitigating global pressures for wage convergence, diverging social policies on these issues can encourage open source development even in developed market economies. Open source development may be popular in Canada, Finland and the Netherlands as relative to their neighbors, for instance, exactly because these countries are in many ways *satellite* economies. Programmers in these nations are likely to be well-aware of international income differentials, and frustrated by market immobility as well. Will a strengthened Euro reduce incentives for open source programming in Europe by reducing the variance in real income across Europe? Will labor mobility across Europe lead to a "convergence" in open source development as the common market becomes a truly continental one? In the case of software piracy, the widespread availability of illegal software drives two conflicting trends. Initially, high levels of piracy should reduce the incentive developers have to build open source alternatives to commercial products. As such, tolerating piracy may become a strategy of self-preservation for certain commercial firms, especially those seeking to establish their products as *de facto* market standards. Tacitly ignoring piracy - for all of its lost revenue - may yet become one of Microsoft's survival tactics, especially in countries like China which have yet to socially institutionalize open-source development networks. Once open source projects are well-established, however, high levels of piracy very clearly undermine commercial software development. It most prominently lowers the opportunity cost of coding free software over commercial applications, and also ushers in the escalating benefits of free software development modeled by Raymond and Ghosh.

Since theory informs us that public goods are always under-provided in market systems (the free-riding problem), the success of packages such as Linux in direct competition with market-driven alternatives offers vindication for policies aimed at improving social welfare through deliberate market distortion. Public policy may prove critical to the continued success of the open source movement – especially in the more subdued equity market following the recent economic downturn in the United States. NASA's support for the MySQL project and the release of the Army-coded GIS package GRASS under an open source license constitute two examples of "soft" support. As market incentives for open source software production shrink, "hard" support for programs such as higher education may be critical factors to the extent that these types of interventions can discourage rent-seeking behavior while encouraging individuals to invest time and effort in the creation of public goods [45]. Any balanced evaluation of the open source model must take these negative social costs into consideration as well.

Given the tentative nature of this research, I feel it vitally necessary to limit the scope of these conclusions. This paper is not intended as a critique of sociological theories about *how* individuals interact once they are in open source communities, but rather as an alternate and hopefully more convincing explanation for *why* developers drift into open source projects and cultures in the first place. Critics may be right to question the universal applicability of these conclusions. It is possible that the true significance of the open source movement is unlikely to be found in the success or failure of any particular project, but in the ephiphenomenal possibilities created the interaction of thousands of smaller, stand-alone programs, application libraries and standards (Raymond's UNIX Gospel). And since important software projects are clearly *not* restricted to the upper left-hand quadrant on which this paper has focussed, it is possible that a very important segment of the puzzle is missing from this analysis [46]. It is impossible to use this evidence to infer about causation in any quadrant but the combination of "Anti-Proprietary" and "Highly-Complex" projects.

But given the tremendous importance of this debate, it is vital not to dismiss without genuine thought and substantial reflection the points political economists bring to bear on this issue. For it is equally plausible that the benefits of the UNIX Gospel are over-exaggerated: possible only in certain systems (such as command-line UNIX desktops) and in limited circumstances (such as when inter-program communication can be handled through character strings). And if the truly important projects in the open source community are those most similar to the Linux and Gnome projects, which are unique in creating (not exploiting) a framework on which further development can be built, the insights political economists can shed on these movements allow for a much more nuanced view of development than is made by advocates of post-scarcity gift cultures. While this view may be less than wholly optimistic in its hopes for man ever

transcending industrial capitalism, it is perhaps reassuring in concurring nonetheless with earlier critics that regardless of how software is produced in the real world, its increasing extensibility seems to be in the public interest and should be encouraged where feasible. End of article

About the Author

David Lancashire is a Ph.D. candidate in the Department of Political Science at the University of California, Berkeley. E-mail: david@socrates.berkeley.edu

Notes

1. Netscape's January 1998 decision to open source the code of its next-generation Web browser was apparently made out of the belief, as CEO Jim Barksdale commented, that Netscape could "ignite the creative energies of the entire Net community" to support its private profitability.

2. It is difficult (perhaps impossible) to conduct an unbiased Simple Random Survey on a subject like open source. One ends up invariably selecting on the dependent variable. Assumptions of what constitutes an open source project selectively influence the population from which sample projects are drawn for analysis.

3. It is foolish to accord non-economic theories of production less serious consideration than those compatible with classical theory, especially when - as in a case like the open source debate - arguments which do not require economic rationality on the part of individuals present complex, sophisticated and plausible hypotheses about production.

4. Raymond completed his undergraduate education in mathematics and philosophy at the University of Pennsylvania.

5. Much of the analytic confusion in the literature stems from the lack of a clear distinction in this regard.

6. That Raymond recognizes the implications of this is clear from "The Magic Cauldron", wherein Raymond writes that "In order to pursue [an economic] analysis without distraction, we'll need to abandon (or at least agree to temporarily ignore) the 'gift culture' level of explanation."

7. Nicholas Bezroukov, "Open Source Software Development as a Special Type of Academic Research," *First Monday,* volume 4, number 10 (October 1999), at http://firstmonday.org/issues/issue4_10/bezroukov/.

8. Marx's Hegelian interpretation of history may make his argument cultural in the broad sense of the term - but he is unquestionably an economic determinist in his expectation that each individual's behavior in markets will be determined by their relationship to the means of production (capital and labor).

9. This probably explains why Raymond is such an intellectual provocateur, and why there have been so many attempts by others to impose "rigor" on his argument according to the dictates of quite divergent schools of thought.

10. The use of the term "gift economy" is profoundly misleading in my opinion - since it proffers a deeply *non-economic* explanation for human behavior, at least as the term *economic* has come to be used in both the popular and academic discourse.

11. It is interesting to note that this tendency to set hackers as a culturally distinct class consistently portrays them as almost diametrically opposed to those figures most associated with conventional economic rationality: business owners, middle-managers, "bean-counters", and the like. "The Story of Mel" may be found online at http://www.pbm.com/~lindahl/mel.html.

12. Quotation taken from *The Jargon File,* a dictionary of computing/coding lexicon available online at http://www.tuxedo.org/~esr/jargon/html/entry/hacker-ethic.html.

13. Turkle, p. 201.

14. Richard Barbrook. "The High-Tech Gift Economy," *First Monday,* volume 3, number 12 (December 1998), at

http://firstmonday.org/issues/issue3_12/barbrook/.

15. This same problem bedevils Raymond's attempt to model open source development "economically" in "The Magic Cauldron." It is significant that the example Raymond uses is of a "J. Random Hacker" forced to decide whether or not to contribute a privately-developed driver to a large existing project rather than a similar hacker facing the question of whether to give away a privately-developed core project (such as an operating system) rather than attempt to commercialize it.

16. Private calculations of author made from source tarball.

17. Microsoft's recent criticism of Open Source seems to hinge on exactly this kind of path-dependent argument about economic production. The claim is that embedding their projects in open source code is likely to cripple the ability of software firms to pursue their own profitability over the long-term.

18. For an explanation of the importance of the "crucial case" concept, a good reference is Harry Eckstein, *Regarding Politics: Essays on Political Theory, Stability, and Change*. Berkeley: University of California Press, 1992, pp.117-177.

19. In other words, this typology is developed to identify projects most representative of "post-scarcity" (altruistic) development.

20. 19 An obvious exception to this case is when simple software algorithms are protected by patent law. As these restrictions affect the development of both *free* and *commercial* software alike, they do not alter the effectiveness of this categorization for analysis of the causal factors of open source production.

21. Critics of KDE focused in particular on its reliance on the QT library, which was not wholly free at the time of initial Gnome development, although it has since been released into the public domain. It seems reasonable to speculate that Gnome development influenced the decision of QT developers to release their software under a general public license.

22. Although the information in this CREDITS file is self-selecting, it seems fair to assume that it nonetheless contains a representative sample of developers important to the project. The precise assumption made here is that any self-selecting tendencies on the part of developers are uncorrelated with their geographic location.

23. It should be noted that this last step (very rarely invoked) should if anything bias any distribution of development towards of the United States, given the "internationalization" of American top-level domains.

24. Per capita figures for *all* contributing countries and all other supporting datasets are available freely upon request from the author. Because these figures include countries which contribute only one developer to a single project, I have chosen to drop them from analysis to avoid relative ranking being distorted by the presence of clear outliers - where support is so low that the assumption that it can be fairly associated with any nation is highly questionable. To note, most of these low-level contributors cluster around the relative bottom of the scale in terms of their relative contribution per capita, although one or two smaller countries like Luxembourg can shift to a high relative position due to their small population base.

25. To maintain methodological consistency and prevent excessive distortion I rely on the Neilsen-Netratings data here. In the cases of Austria, Hungary, and Spain (countries for which Neilsen-Netratings data is unavailable) I have relied on the most recent estimates available from NUA, at http://www.nua.ie.

26. Nakagawa, 2001. "From Local to Global," p. 108.

27. Linus Torvalds published the first announcement of Linux, as well as a request for help developing it, on an English-language USENET group.

28. Milton Friedman. *Essays in Positive Economics*. Chicago: University of Chicago Press, 1953, p. 14.

29. National level unemployment figures do not appear to be strongly if at all correlated with the vibrancy of open source communities. This is to be expected given an assumption of asset-specificity in human capital - in which case the performance of non IT sectors will significantly influence the national economic data, but may inaccurately represent developments in the country's high-tech sector.

30. The focus on sectors instead of factors implies a belief in asset-specificity. This implies that wages for programmers

will rise because (at least in the short term) there are barriers to entry in the labor market. The hypothesized reduction in open source production is again contingent on holding all other factors constant.

31. One explanation emphasizes the liberalization of the American equity market and the growth in day-trading throughout the 1990s. Others point to the structural advantages of equity-driven versus bank-driven financial systems for financing high-risk ventures. I am personally partial to an explanation linking soaring levels of high-risk portfolio investment in the United States to the fallout of the East Asian economic crisis of 1997/1998 and the repatriation of funds invested abroad to American markets. Regardless, this is a puzzle into which further research will be necessary to introduce clarity.

32. This last connection is not automatic. An argument can be made that historic "revolutions" in industrial processes privileged capital over labor, creating downward pressure on wages (if not on real income) since the skills necessary to operate new machinery could be gained relatively easily while rising unemployment created a broad pool of competitive workers.

33. Price Waterhouse. "The Contribution of the Packaged Software Industry to the European Economies" (May 1998).

34. Freeman and William Aspray, 1999. *The Supply of Information Technology Workers in the United States*.

35. John Maynard Keynes. *The Economic Consequences of the Peace*. London: Macmillan, 1919.

36. Nakagawa, 2001. "From Local to Global," p. 109.

37. *Ibid.,* p. 108.

38. "Forecasting a Robust Future," Nathan Associates (1999).

39. This competitive advantage does not necessarily transfer into non software-driven industries, as European dominance of the wireless services market seems to suggest.

40. This is a key point made in Lerner and Tirole (2000).

41. Hatton and Williamson, "International Migration 1850-1939: An economic survey," *Migration and the International Labor Market* (1994), p. 8.

42. Hatton and Williamson, "International Migration 1850-1939: An economic survey," *Migration and the International Labor Market* (1994).

43. The history of TCP/IP is one very prominent exception.

44. "Well, I've been employed by the University of Helsinki, and they've been perfectly happy to keep me employed and doing Linux. Doing Linux isn't officially part of my job description, but that's what I've been doing, and they obviously know and support that - in a sense I do get my pizzas paid for by Linux indirectly." See "FM Interviews: Linus Torvalds - What Motivates Free Software Developers?" *First Monday,* volume 3, number 3 (March 1998), at http://firstmonday.org/issues/issue3_3/torvalds/.

45. Further research on patterns of public investment are needed. There are economic inefficiencies associated with certain types of subsidization which may or may not dominate the benefits created by them in the real world.

46. One problem frustrating the analysis of developer distribution in smaller software programs is the tendency of these projects not to release detailed contact information, or to rely on commercial software managers like Sourceforge which mask developer information.

Selected Bibliography

Hillary Bays and Miranda Mowbray, 1999. "Cookies, Gift-Giving and the Internet," *First Monday,* volume 4, number 11 (November), at http://firstmonday.org/issues/issue4_11/bays/.

Hillary Bays, 1998. "Framing and Face in Internet Exchanges," *Linguistik Online,* (January), at http://viadrina.euv-

frankfurt-o.de/~wjournal/bays.htm.

Richard Barbrook, 1998. "The High-Tech Gift Economy," *First Monday,* volume 3, number 12 (December), at http://firstmonday.org/issues/issue3_12/barbrook/.

Nicholas Bezroukov, 1999a. "Open Source Software Development as a Special Type of Academic Research (Critique of Vulgar Raymondism)," *First Monday,* volume 4, number 10 (October), at http://firstmonday.org/issues/issue4_10/bezroukov/.

Nicholas Bezroukov, 1999b. "A Second Look at the Cathedral and the Bazaar," *First Monday,* volume 4, number 12 (December), at http://firstmonday.org/issues/issue4_12/bezroukov/.

Harry Eckstein, 1992. *Regarding Politics: Essays on Political Theory, Stability, and Change*. Berkeley: University of California Press.

European Telework Online, 2001. "The e-commerce gap: Europe-USA eight year comparison," at http://www.eto.org.uk/eustats/graphs/93-99.htm.

Peter Freeman and William Aspray, 1999. *The Supply of Information Technology Workers in the United States*. Washington, D.C.: Computing Research Association, at http://cra.org/reports/wits/cra.wits.html.

Milton Friedman, 1953. *Essays in Positive Economics*. Chicago: University of Chicago Press.

Rishab Aiyer Ghosh, 1998. "Cooking-Pot Markets: An Economic Model for the Trade in Free Goods and Services Over the Internet," *First Monday,* volume 3, number 3 (March), at http://firstmonday.org/issues/issue3_3/ghosh/.

Timothy J. Hatton and Jeffrey G. Williamson, 1994. "International Migration 1850-1939: An Economic Survey," In: Timothy J. Hatton and Jeffrey G. Williamson (editors). *Migration and the International Labor Market, 1850-1939*. London: Routledge.

Pekka Himanen, 2001. *The Hacker Ethic, and the Spirit of the Information Age*. New York: Random House.

John Maynard Keynes, 1919. *The Economic Consequences of the Peace*. London: Macmillan.

Ko Kuwabara, 2000. "Linux: A Bazaar at the Edge of Chaos," *First Monday,* volume 5, number 3 (March), at http://firstmonday.org/issues/issue5_3/kuwabara/.

Josh Lerner and Jean Tirole, 2000. "The Simple Economics of Open Source," National Bureau of Economic Research (NBER) Working Paper 7600 (March), at http://papers.nber.org/.

Trevor Nakagawa, 2001. "From Local to Global: European Enterprise Software Strategies in Asia," In: Vinod K. Aggarwal (editor). *Winning in Asia, European Style*. New York: Palgrave, pp. 107-158.

Nathan Associates, 1999. *Forecasting a Robust Future*. Arlington, Va.: Nathan Associates.

Douglass C. North, 1990. *Institutions, Institutional Change, and Economic Performance*. New York: Cambridge University Press.

Price Waterhouse, 1998. "The Contribution of the Packaged Software Industry to the European Economies: A Study by Price Waterhouse Commissioned by the Business Software Alliance," (May), at http://www.bsa.org/europe-eng/globallib/BSA_Piracy_Green_Paper_Commen_2.pdf.

Eric S. Raymond, 1998a. "The Cathedral and the Bazaar," *First Monday,* volume 3, number 3 (March), at http://www.firstmonday.org/issues/issue3_3/raymond/.

Eric S. Raymond, 1998b. "Homesteading the Noosphere," *First Monday,* volume 3, number 10 (October), at http://www.firstmonday.org/issues/issue3_10/raymond/.

Eric S. Raymond, 1999a. "The Magic Cauldron," privately published at http://www.tuxedo.org/~esr/writings.

Eric S. Raymond, 1999b. "A Response to Eric Bezroukov," *First Monday,* volume 4, number 11 (November), at http://www.firstmonday.org/issues/issue4_11/raymond/.

Joseph Schumpeter, 1950. *Capitalism, Socialism and Democracy*. New York: Harper.

Giancarlo Succi, James Paulson and Armin Eberlein, 2001. "Preliminary Results from an Empirical Study on the Growth of Open Source and Commercial Software Products," paper presented at the International Conference on Software Engineering 2001, Toronto, Ontario, Canada, (12-19 May), at http://sern.ucalgary.ca/~paulson/publications/icse2001_paper_paulson.pdf/

Linus Torvalds, 1998. "FM Interviews: Linus Torvalds - What Motivates Free Software Developers?" *First Monday,* volume 3, number 3 (March 1998), at http://firstmonday.org/issues/issue3_3/torvalds/.

Sherry Turkle, 1984. *The Second Self: Computers and the Human Spirit*. New York: Simon and Schuster.

Steven Weber, 2000. "The Political Economy of Open Source Software," (Berkeley Roundtable on the International Economy) BRIE Working Paper 140 (June), at http://brie.berkeley.edu/~briewww/pubs/wp/wp140.pdf.

Appendix: Supporting Data

Table 2: Complete Data Table

Country	Linux	Gnome	Population*	Nielsen-Netratings**	Nua***	Linux PC	Gnome PC	Combi PC
Finland	9	0	5175783	1977637	0	575087	0	575,0
Romania	1	0	22364022	0	600000	22364022	0	22364
USA	132	31	278058881	165180807	0	2106506.674	8969641.323	170588
Ireland	2	0	3840838	1250404	0	1920419	0	19204
United Kingdom	31	7	59647790	23870341	0	1924122.258	8521112.857	156967
Czech Republic	10	0	10264212	0	1100000	1026421.2	0	10264
Belgium	3	0	10258762	0	2700000	3419587.333	0	341958
Canada	20	2	31592805	14445047	0	1579640.25	15796402.5	143603
Italy	4	4	57679825	18697197	0	14419956.25	14419956.25	720997
France	9	1	59551227	11107974	0	6616803	59551227	59551
Denmark	6	2	5352815	2930032	0	892135.8333	2676407.5	669101
Germany	64	5	83029536	27914911	0	1297336.5	16605907.2	120332
Australia	19	3	19357594	9674157	0	1018820.737	6452531.333	879890.
Poland	2	0	38633912	0	2800000	19316956	0	19316

Brazil	9	0	174468575	0	9840000	19385397.22	0	1938539
Netherlands	18	3	15981472	8671316	0	887859.5556	5327157.333	761022.
Sweden	8	10	8875053	5543193	0	1109381.625	887505.3	49305
Norway	2	1	4503440	2452772	0	2251720	4503440	150114
Hungary	2	3	10106017	0	730000	5053008.5	3368672.333	202120
Taiwan	1	0	22370461	0	6400000	22370461	0	22370
Argentina	1	1	37384816	0	800000	37384816	37384816	18692
Ukraine	1	0	48760474	0	200000	48760474	0	48760
South Africa	1	0	43586097	0	1820000	43586097	0	43586
Mexico	1	3	101879171	0	1820000	101879171	33959723.67	2546979
Hong Kong	1	1	7210505	3935769	0	7210505	7210505	360525
Bulgaria	1	0	7707495	0	200000	7707495	0	77074
Croatia	1	0	4334142	0	100000	4334142	0	43341
Russia	1	0	145470197	0	9200000	145470197	0	145470
Austria	3	2	8150835	0	3000000	2716945	4075417.5	16301
Spain	2	3	40037995	0	5490000	20018997.5	13345998.33	80075
New Zealand	1	0	3864129	1747203	0	3864129	0	38641
Switzerland	1	0	7283274	0	2400000	7283274	0	72832
Japan	1	0	126771662	42573749	0	126771662	0	126771
Egypt	0	1	69536644	0	440000	0	69536644	69536
India	0	1	1029991145	0	4500000	0	1029991145	102999
Portugal	0	3	10066253	0	700000	0	3355417.667	3355417
Slovenia	0	1	1930132	0	460000	0	1930132	19301
Turkey	0	1	66493970	0	2000000	0	66493970	66493

* Population data from the IDB; ** Neilsen Netratings; *** Latest data available from NUA

Table 3: Developers >= 2 for both Projects

Country	Linux	Rank	Gnome	Rank	Population*	Nielsen-Netratings**	Nua***	Linux PC	Rank
USA	132	1	31	1	278058881	165180807	0	2106506.674	8
United Kingdom	31	3	7	3	59647790	23870341	0	1924122.258	7
Canada	20	4	2	10	31592805	14445047	0	1579640.25	6
Italy	4	9	4	5	57679825	18697197	0	14419956.25	11
Denmark	6	8	2	10	5352815	2930032	0	892135.8333	2
Germany	64	2	5	4	83029536	27914911	0	1297336.5	5
Australia	19	5	3	6	19357594	9674157	0	1018820.737	3
Netherlands	18	6	3	6	15981472	8671316	0	887859.5556	1
Sweden	8	7	10	2	8875053	5543193	0	1109381.625	4
Hungary	2	11	3	6	10106017	0	730000	5053008.5	10
Austria	3	10	2	10	8150835	0	3000000	2716945	9
Spain	2	11	3	6	40037995	0	5490000	20018997.5	12

* Population data from the IDB; ** Nielsen Netratings; *** Latest data available from NUA

Table 4: Nielsen-Netratings Countries

Country	Linux	Rank	Gnome	Rank	Population*	Nielsen-Netratings**	Nua***	Linux PC	Rank
Finland	9	7	0	13	5175783	1977637	0	575087	1
USA	132	1	31	1	278058881	165180807	0	2106506.674	10
Ireland	2	12	0	13	3840838	1250404	0	1920419	8
United Kingdom	31	3	7	3	59647790	23870341	0	1924122.258	9
Canada	20	4	2	8	31592805	14445047	0	1579640.25	7
Italy	4	11	4	5	57679825	18697197	0	14419956.25	16
France	9	7	1	10	59551227	11107974	0	6616803	13

Denmark	6	10	2	8	5352815	2930032	0	892135.8333	3
Germany	64	2	5	4	83029536	27914911	0	1297336.5	6
Australia	19	5	3	6	19357594	9674157	0	1018820.737	4
Netherlands	18	6	3	6	15981472	8671316	0	887859.5556	2
Sweden	8	9	10	2	8875053	5543193	0	1109381.625	5
Norway	2	12	1	10	4503440	2452772	0	2251720	11
Hong Kong	1	14	1	10	7210505	3935769	0	7210505	14
New Zealand	1	14	0	13	3864129	1747203	0	3864129	12
Japan	1	14	0	13	126771662	42573749	0	126771662	15

* Population data from the IDB; ** Nielsen Netratings; *** Latest data available from NUA

Table 5: Linux Developers Only

Country	Linux	Population*	Nielsen-Netratings**	Nua***	Linux PC	Rank	Linux POC	Rank
Finland	9	5175783	1977637	0	575087	1	219737.4444	6
Romania	1	22364022	0	600000	22364022	26	600000	12
USA	132	278058881	165180807	0	2106506.674	11	1251369.75	22
Ireland	2	3840838	1250404	0	1920419	9	625202	13
United Kingdom	31	59647790	23870341	0	1924122.258	10	770011	16
Czech Republic	10	10264212	0	1100000	1026421.2	5	110000	2
Belgium	3	10258762	0	2700000	3419587.333	14	900000	18
Canada	20	31592805	14445047	0	1579640.25	8	722252.35	15
Italy	4	57679825	18697197	0	14419956.25	22	4674299.25	30
France	9	59551227	11107974	0	6616803	18	1234219.333	3
Denmark	6	5352815	2930032	0	892135.8333	3	488338.6667	10
Germany	64	83029536	27914911	0	1297336.5	7	436170.4844	8

Australia	19	19357594	9674157	0	1018820.737	4	509166.1579	11
Poland	2	38633912	0	2800000	19316956	23	1400000	23
Brazil	9	174468575	0	9840000	19385397.22	24	1093333.333	20
Netherlands	18	15981472	8671316	0	887859.5556	2	481739.7778	9
Sweden	8	8875053	5543193	0	1109381.625	6	692899.125	14
Norway	2	4503440	2452772	0	2251720	12	1226386	21
Hungary	2	10106017	0	730000	5053008.5	17	365000	7
Taiwan	1	22370461	0	6400000	22370461	27	6400000	31
Argentina	1	37384816	0	800000	37384816	28	800000	17
Ukraine	1	48760474	0	200000	48760474	30	200000	4
South Africa	1	43586097	0	1820000	43586097	29	1820000	25
Mexico	1	101879171	0	1820000	101879171	31	1820000	25
Hong Kong	1	7210505	3935769	0	7210505	19	3935769	29
Bulgaria	1	7707495	0	200000	7707495	21	200000	4
Croatia	1	4334142	0	100000	4334142	16	100000	1
Russia	1	145470197	0	9200000	145470197	32	9200000	32
Austria	3	8150835	0	3000000	2716945	13	1000000	19
Spain	2	40037995	0	5490000	20018997.5	25	2745000	28
New Zealand	1	3864129	1747203	0	3864129	15	1747203	24
Switzerland	1	7283274	0	2400000	7283274	20	2400000	27
Japan	1	126771662	42573749	0	126771662	33	42573749	33

* Population data from the IDB; ** Nielsen Netratings; *** Latest data available from NUA

Table 6: Gnome Developers Only

Country	Gnome	Rank	Population*	Nielsen-Netratings**	Nua***	Gnome PC	Rank	Gnome POC
USA	31	1	278058881	165180807	0	8969641.323	12	5328413.129

Argentina	1	15	37384816	0	800000	37384816	18	800000
United Kingdom	7	3	59647790	23870341	0	8521112.857	11	3410048.714
Mexico	3	3	101879171	0	1820000	33959723.67	17	606666.6667
Hong Kong	1	15	7210505	3935769	0	7210505	10	3935769
Canada	2	12	31592805	14445047	0	15796402.5	15	7222523.5
Italy	4	5	57679825	18697197	0	14419956.25	14	4674299.25
France	1	15	59551227	11107974	0	59551227	19	11107974
Denmark	2	12	5352815	2930032	0	2676407.5	3	1465016
Germany	5	4	83029536	27914911	0	16605907.2	16	5582982.2
Australia	3	3	19357594	9674157	0	6452531.333	9	3224719
Austria	2	12	8150835	0	3000000	4075417.5	6	1500000
Spain	3	3	40037995	0	5490000	13345998.33	13	1830000
Netherlands	3	3	15981472	8671316	0	5327157.333	8	2890438.667
Sweden	10	2	8875053	5543193	0	887505.3	1	554319.3
Norway	1	15	4503440	2452772	0	4503440	7	2452772
Hungary	3	3	10106017	0	730000	3368672.333	5	243333.3333
Egypt	1	15	69536644	0	440000	69536644	21	440000
India	1	15	1029991145	0	4500000	1029991145	22	4500000
Portugal	3	3	10066253	0	700000	3355417.667	4	233333.3333
Slovenia	1	15	1930132	0	460000	1930132	2	460000
Turkey	1	15	66493970	0	2000000	66493970	20	2000000

* Population data from the IDB; ** Nielsen Netratings; *** Latest data available from NUA

Editorial history

Paper received 5 November 2001; accepted 19 November 2001.

Contents Index

The Fading Altruism of Open Source Development by David Lancashire
First Monday, volume 6, number 12 (December 2001),
URL: http://firstmonday.org/issues/issue6_12/lancashire/index.html