What have The Romans IS academics ever done for us? The Lessons of Open-Source

Antony Bryant
Leeds Metropolitan University

Follow this and additional works at: http://aisel.aisnet.org/icis2006

Recommended Citation
http://aisel.aisnet.org/icis2006/102

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2006 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
‘WHAT HAVE THE ROMANS IS ACADEMICS EVER DONE FOR US?’

THE LESSONS OF OPEN-SOURCE

IT for Underserved Communities

Antony Bryant
Leeds Metropolitan University
a.bryant@leedsmet.ac.uk

Abstract

This paper seeks to develop the motivations and aspirations underlying the primary theme for ICIS 2006 – ‘IT for Underserved Communities’. In so doing the case is made that those keen to mobilize and harness the emancipatory and empowering potential of Information & Communications Technology for community-based projects should consider that the very existence of this technology opens up alternative models of co-operation and collaboration. These models themselves provide the basis for breaking away from ‘traditional’ command-and-control models of management and co-ordination; allowing participants, or potential participants, to co-ordinate their efforts along the lines exemplified by the open-source software movement and the contributors to Wikipedia: Models of co-ordination that ought not to work, but appear to do so.

Keywords: Open-source, Linux, Wikipedia, Wiki movement, community, globalization, IS academics

Introduction

A few years ago Frank Land laid down a challenge to the IS academic community, via ISWORLD. He noted that ‘the IS academic community…has not itself initiated new uses of IS. We note that the IS community is very good at scrambling aboard the latest bandwagon, as often engaging in the hype rather than being properly critical….Few of the bandwagons survive more than a few years. Where is the work in our community which yields new ideas on how IS can serve mankind? Am I being unfair?’ (Land 2004). Or as Reg from Monty Python’s Life of Brian might have phrased it – ‘What have IS academics ever done for us?’

Land’s challenge did not evoke any startlingly significant responses that might have tempered his tone of disappointment and frustration. Yet perhaps those seeking to rise to his challenge were looking in the wrong place, trying to find an example of a specific application or technical innovation, or were simply too busy jumping aboard the latest bandwagon. In any case within the field of IS we are far more used to taking than giving – or perhaps to put it more kindly, borrowing rather than lending. We use terms such as information and system often with little regard to their earlier uses. We are complicit with computer engineers and software engineers, acting in concert, as some of the most experienced and successful of conceptual magpies. We borrow terms from other realms, and

1 Anyone not familiar with the film Life of Brian can find out more at the Wikipedia site (warning – it contains a spoiler) http://en.wikipedia.org/wiki/Monty_Python's_Life_of_Brian. The text from the scene alluded to here can be found at http://www.anenglishmanscastle.com/archives/001524.html – the ironic finale: ‘All right, but apart from the sanitation, the medicine, education, wine, public order, irrigation, roads, a fresh water system, and public health, what have the Romans ever done for us?’
apply them to different contexts and with different references. This is often accomplished so successfully that the earlier meaning of concepts is effaced by the new ones; examples include input, editor, desktop, program and icon.2

The term engineer itself – and a whole host of associated terms such as requirements, specification, development project, life-cycle, design, construction, maintenance, and so on – have been lifted from the domain of general engineering and applied to aspects of systems and software development. In some cases the application has been illuminating and beneficial, but there are others where it has proved far more ambivalent, if not a downright hindrance. Moreover, the term engineering itself when applied to systems and software appears to have become part of the problem rather than offering a solution. F.P. Brooks (1986) pointed out many years ago that software was better seen as nurtured and cultivated rather than built. He argued that the construction metaphor had been an important conceptual breakthrough when first used; prior to that software had been thought of as something one wrote, i.e., similar in many respects to a lone author working on a novel. (It should be noted that this is disputed by some pioneers such as Land, who argues that even the earliest commercial systems were developed on a collaborative basis – and this was an essential feature leading to their success – see Glass 1998.) Whatever its origins, the construction metaphor brought with it a number of issues and concerns that previously had perhaps not been at the forefront of software developers’ priorities. As Brooks summarized it:

I still remember the jolt I felt in 1958 when I first heard a friend talk about building a program, as opposed to writing one. In a flash he broadened my whole view of the software process. The metaphor shift was powerful, and accurate. Today we understand how like other building processes the construction of software is, and we freely use other elements of the metaphor, such as specifications, assembly of components, and scaffolding. (1986, p.200)

In place of the engineering or construction metaphor, Brooks proposed the metaphor of nurture and cultivation. More than that he offered an outline of how the development process might be seen as an incremental strategy leading from the simple to the complex.

Let us turn [to] nature and study complexity in living things, instead of just the dead works of man. Here we find constructs whose complexities thrill us with awe. The brain alone is intricate beyond mapping, powerful beyond imitation, rich in diversity, self-protecting, and self-renewing. The secret is that it is grown, not built.3

So it must be with our software-systems. Some years ago Harlan Mills proposed that any software system should be grown by incremental development.4 That is, the system should first be made to run, even if it does nothing useful except call the proper set of dummy subprograms. Then, bit by bit, it should be fleshed out, with the subprograms in turn being developed--into actions or calls to empty stubs in the level below. (1986, p.201)

In these writings Brooks offered his readers a glimpse of a different model of software development, albeit one based more on an intuition than a specification. Yet at the time that he wrote this, there was already an important development in train. The 1970s had witnessed the separation of hardware from software, to the extent that operating software was now seen as commercially confidential, and those wishing to have access to the source code had to sign non-disclosure agreements (Levy 1984). At MIT, Richard Stallman, frustrated at being prevented from making even simple modifications to software, sought to establish a counter-movement to proprietary software. Thus was born the Free Software Foundation [FSF], the General Public Licence [GPL] and the concept of ‘copyleft’ (see Levy 1984) and what is now referred to as the open-source movement. Here perhaps we have a substantive example with which to respond to Land’s challenge, one that might even be seen as ‘serving mankind’.

The Open-Source Movement – ‘The Cathedral and the Bazaar’

The open-source movement can trace its origins to, and look to its founding figure in Richard Stallman and the establishment of the Free Software Foundation in 1985. But its real breakthrough came in the 1990s with the work

---

2 The history of such linguistic transformations dates back at least to the 19th century when the term computer referred to a person not a machine – and in similar fashion we may well have to remind our grandchildren about the original meaning of the term printer.

3 It might have been better if this had read – ‘The secret is that it has grown, not been built.’

associated with the Linux operating system, inspired by Linus Torvalds, although Stallman is still regarded as the ‘hacker-in-chief’, sufficiently famous to be known simply as RMS. Moreover if Torvalds is the key figure associated with the true burgeoning of the open-source model of development and collaboration, then Eric Raymond’s writings can be regarded as the manifesto of the movement, in particular his paper which introduced the contrast between the cathedral and the bazaar (2001).

For Raymond the cathedral model is one relying on the efforts of ‘individual wizards or small bands of mages working in splendid isolation’; the entire edifice or product needs to be completed and fully guaranteed or secured prior to its ‘release’. The alternative ‘resembles a great babbling bazaar of differing agendas and approaches (aptly symbolized by the Linux archive sites, who’d take submissions from anyone) out of which a coherent and stable system could seemingly emerge only by a succession of miracles’ (2001, pp.21-2).

Raymond’s argument centres on software development, more specifically on software debugging. Yet it appears to offer a rich and valuable series of insights that reach well beyond this distinct domain, offering an archetype not only for systems development, but for a far wider form of community-based initiatives. The cathedral model relies on a small group of proficient developers working in splendid isolation, only releasing their software to users after extensive and thorough testing – all of which takes time and effort. In stark contrast stands the bazaar-like model, whereby disparate groups and individuals with differing agendas and approaches somehow produce a coherent and stable outcome. The Linux philosophy is encapsulated in Linus Torvalds’ philosophy as stated by Raymond – ‘release early and release often; delegate everything you can, be open to the point of promiscuity’. The result ought to be chaotic and anarchic, a configurational nightmare with contending versions of software proliferating to the consternation of developers and the despair of users and customers. Yet precisely the opposite has occurred. Linux has survived, thrived and continues to flourish. Moreover the model has been adopted by others, and Raymond’s paper from 1997 – later republished as the title chapter in his book (2001) – can be taken as a manifesto for this type of co-operative venture, although as will be explained in what follows, he crucially undersells his own analysis.  

Volunteers and Collaboration

At a more general level the lesson of the bazaar approach is that complex development projects reach effective and sustained outcomes when involvement is voluntary and collaborative. There is control and management, but it is enacted in a distributed and semi-autonomous fashion. People take on tasks and responsibilities because they feel motivated to contribute and exchange their views, their ideas and their efforts. The development of open-source software – specifically Linux – has come about via a model based on several principles that at first sight ought not to prove effective or successful, but they have, and they do. The question at this point is can such principles of operation be abstracted from the realm of software development and applied elsewhere? Are there lessons to be learned not grounded in the IT, but rather in this model of co-operation?

Raymond offers a number of maxims or aphorisms – 19 in all – to summarize this approach. Many of them can be extended and help to develop an understanding of how the open-source model might offer a far wider series of insights and exemplars, as well as a substantive response to Land’s challenge. Raymond’s first maxim is: ‘Every good work of software starts by scratching a developer’s personal itch.’ The key issue here, and one that comes through very clearly in Raymond’s description of his own participation in the open-source community, is that involvement must at least owe some of its initial impetus to the motivation and enthusiasm of the participant. This is not to subscribe to any simplistic appropriation of the concept of volunteering, a misappropriation that is often justified with reference to Drucker’s purported maxim that ‘in the knowledge society we are all volunteers’. It should be obvious that most of the working population do not engage in their particular employment voluntarily. The largest private employer in the US is Wal-Mart, and few of their 1.2 million employees would consider

---

5 The distinctions between FSF and Linux/open-source are significant – but at this juncture it is more important to focus on their general similarities, and to acknowledge the precedence of FSF.
6 The term ‘hacker’ is used here in its original meaning of someone who hacks code – i.e., someone who produces and modifies computer software. So hacker is a term of approval, indicating that the person is skilled and experienced at the complex task of software development. Moreover it is a term that has to be bestowed by one’s peers rather than claimed for oneself.
7 Recent analyses of the Linux movement bring some of the received wisdom into question – particularly the primacy of Torvalds himself, and the hierarchy that emanates from him – see Weber 2005. In general, however, the key points about the model and its ramifications and exemplary function retain their validity.
themselves ‘volunteers’, although the company itself uses the term ‘associate’ for all its employees whether CEO or checkout assistant. Drucker’s maxim has a crucial corollary: ‘Everyone in the knowledge economy is a volunteer, but we have only trained our managers to manage conscripts’. For our present purposes the key points were developed by Drucker himself in an article for Forbes Magazine in the late 1990s.

What motivates workers -- especially knowledge workers -- is what motivates volunteers. Volunteers, we know, have to get more satisfaction from their work than paid employees precisely because they do not get a pay check. They need, above all, challenge. They need to know the organization's mission and to believe in it. They need continuous training. They need to see results. … Implicit in this is that employees have to be managed as associates, partners -- and not in name only. The definition of a partnership is that all partners are equal. It is also the definition of a partnership that partners cannot be ordered. They have to be persuaded. (Drucker, 1998, stress added)

Drucker makes the case that concepts such as partnership, challenge and the like, ought to motivate workers, but it is all too obvious that this has little relevance to the vast majority of people’s experience. Moreover statements such as this lend themselves too easily to superficial incorporation – despite Drucker’s caveat ‘and not in name only’. Wal-Mart’s use of the term ‘associates’ might have been derived from Drucker; whether their employment practices and employee development policies are similarly inspired and grounded is far more doubtful. Furthermore, many of the current discussions about ‘knowledge management’, ‘knowledge workers’, and the like within the IS-world – and on ISWORLD – often take a similarly partial or naïve position. This is not to deny the existence of more critical voices within the IS academy; on the contrary the central argument of this paper is that there is a potent and pragmatic contribution that can be further developed emanating from within this IS-initiated orientation.

Raymond’s description of the open-source model demonstrates many of the features evoked by Drucker’s characterization of partnership and volunteers. The participants are motivated by the challenge, and the capacity to act as partners. The key concern is how to ensure that such motivations and enthusiasms are nurtured to materialize into specific projects, objectives, and achievements. The technology itself, in the form of ICT, provides a forum and a mechanism for this. Indeed Raymond’s entire discussion of the development of the open-source movement is premised on ICT, specifically the Internet, as a constitutive and indispensable feature. Without such resilient, extensive, and virtually effortless communication, the open-source community simply would not have been able to develop as far as it did. It is no coincidence that Linux burst upon the scene in the early 1990s, precisely the point at which the Internet, as a key component of everyday life for a significant and rapidly growing proportion of people, was taking up its central role as the communications technology par excellence. A prime distinction between the cathedral and the bazaar is the difference between ‘individual wizards…working in splendid isolation’, and ‘a great babbling bazaar of differing agendas and approaches’. The former requires little or no communication and cooperation, the latter demands it as perpetually available; and as shall be seen, the form taken by this combination of communication and co-operation is critical.

Raymond recognizes the central and vital role of the Internet, particularly in providing a safeguard and counter-force to Brooks’ dictum that ‘adding more people to a project that is running late, makes it later still’. Brooks was making the point that the addition of more people might seem to be a good idea, on the basis that ‘many hands make light work’; but this is more than outweighed by the necessity to spend effort bringing the neophytes up to speed, and then coping with the increased communication demands – ‘too many cooks spoil the broth’. Raymond counters what is now called Brooks’ Law as follows: ‘Provided the development coordinator has a communications medium at least as good as the Internet, and knows how to lead without coercion, many heads are inevitably better than one.’ [#19]

---

8 For an interesting series of statistics about Wal-Mart see www.pbs.org/wgbh/pages/frontline/shows/walmart/secrets/stats.html
9 The quote is attributed to Drucker by Snowden, although it is not found in any of Drucker’s writings – stress added – see http://203.81.46.67/pcio/ns/fm/dldlookup/1F4F352F80DADF4F448256B3E0024C456?OpenDocument
10 The importance of the Internet is not that it allows individuals to talk to each other on an individual basis, but that it affords a forum for exchange and co-operation; with both asynchronous and synchronous interactions. There is still an issue with regard to the point at which ‘many’ becomes ‘too many’, as well as the role of gate-keepers and controllers – this is taken up to some extent in a later section when the Wiki movement is discussed.
So far so good: Volunteers are better than conscripts, but note that more is at stake here. There is a need for people to feel motivated and enthused by some concept of mission or purpose. In the commercial world this is often taken to mean that any venture must have a Mission Statement – capital ‘M’, capital ‘S’. A quick search on the Internet demonstrates that there are several packages and tutorial outlines for how best to develop a mission statement. There is even a mission statement generator, found on the Dilbert.com Web site.11 But mission statements are rarely the result of volunteers giving succinct expression to their overwhelming motivations and inspirations. They are usually imposed from above, possibly derived from the CEO’s ‘personal itch’, but also likely to have been generated with regard more to impact, and enhancement of brand recognition, than as an expression of widely felt and deeply held commitment.

In any case mission statements will be neither necessary nor sufficient; at best they may serve as a rallying call or provide enhanced visibility and recognition. More critical is the extent to which people feel that they are being treated as associates, open to persuasion and believing in the mission as something that has what Raymond terms ‘plausible promise’. He develops this theme in some of his other maxims, including the following: ‘Treating your users as co-developers is your least-hassle route to rapid code improvement and effective debugging.’ [§6] ‘If you treat your beta-testers as if they’re your most valuable resource, they will respond by becoming your most valuable resource.’ [§10]

The Concept of ‘Community’

It should be noted that Raymond does not use the term community in any of his maxims, although elsewhere in his article he uses the term frequently – e.g. Linux community, co-developer community, open-source community. His preferred terms in his aphorisms are far more specific to the task in hand, co-ordination for software development, hence co-developers, beta-testers, users, and so on. In this way he makes no claims for being in the business of community building, and simultaneously relates his ideas strictly to the matter-in-hand. This neatly avoids the issue of any simple recourse to the concept of some imagined community. There is a collectivity or sodality of sorts involved here, one intimately associated with a specific task or project, and it may or may not have some more durable existence beyond those confines. The key point is that there needs to be a participative orientation from all sides, which is what Raymond stresses when noting the important roles played by co-developers and beta-testers.

The theme for ICIS 2006 is ‘IT for Underserved Communities’. A laudable topic, and one that immediately and necessarily draws attention to the ways in which IT might best be used to remedy the inequities and injustices that lead to some being better served than others. A similar note was struck by the IFRC12 World Disasters Report 2005

Looking back over the events of 2004, it is striking how many of the year’s disasters could have been avoided with better information and communication. For tens of thousands of people, disaster arrived suddenly, unannounced. IFRC World Disasters Report 2005: Introduction

The IFRC Report drew attention to the informational and communicational aspects of disaster relief, planning and the like. Its title, ‘Information: a life-saving resource’, might be taken to imply that what is needed is a strategy to ensure that the Internet and accompanying technologies reach across the world in ever more pervasive and accessible ways; then everything will be – if not perfect – then certainly better: And the theme for ICIS 2006 might be understood in the same fashion. Yet the authors of the report were at pains to point out that their concerns do not begin and end with the Internet; the type of information, its form, and the ways in which it is gathered and broadcast are of paramount importance. They noted that in many instances the formal, official, centrally-controlled information channels proved far less effective than the informal, dispersed, locally-generated ones. They also stressed that education, preparedness, trust and motivation are key factors affecting the ways in which people

11 Anyone keen to develop a mission statement can download a package from http://www.sound-business-practices.com/rationale/mission-statements.htm. The Web site states that ‘Our Mission Statements Formulation Package contains several example Mission Statements, is easily customizable in almost any word processing software, and most of all, there is no waiting!’ The Dilbert mission statement generator, complete with ready-split infinitives can be found at http://www.dilbert.com/comics/dilbert/games/career/bin/ms.cgi
12 International Federation of Red Cross & Red Crescent Societies
13 This may appear at first sight a strangely formulated observation; after all disasters such as the Asian Tsunami of December 2004 were hardly avoidable. The key point is that the impact and aftermath of such devastations and upheavals can be significantly ameliorated in areas where people are especially vulnerable with more effective use of information and communication.
communicate with and understand one another – regardless of the technology. Likewise the theme for ICIS 2006 clearly moves well beyond any narrow understanding of IT and of Underserved Communities – but this expanded perspective needs to be articulated, and this paper seeks to contribute to this while characterizing the important ramifications of the lessons learned from the open-source experience to date.

The authors of the IFRC report understand information and communication from an anthropological or sociological perspective as opposed to a technical one. So ICT\(^{14}\) is seen as something that provides new ways in which people’s information-seeking behaviours and meaning-sustaining interactions can be enacted as social achievements, as opposed to seeing ICT purely as a series of technological artefacts, and hence viewing information and communication as essentially processes of transfer, storage and retrieval. This was first characterized by Feldman and March (1981) in their distinction between information as signal and symbol. The IFRC Report offers several examples that clearly belong to the information-as-symbol perspective; moreover where informal, locally-based information proved to be far superior – on any measure – to that emanating from official sources.

Vijayakumar Gunasekaran, based in Singapore, heard of the tsunami’s devastating impact on the radio early on the morning of 26 December. He phoned a warning through to his family in Nallavadu on the eastern coast of India, in time for villagers to evacuate all 3,630 residents to safety.

In Tamil Nadu, the Indian state hit hardest by the tsunami, local civil society groups formed a coordination cell to capture people’s priorities across 100 disaster-struck villages and report back on what aid officials were planning. Maintaining communication with affected people is a crucial way in which aid organizations can promote transparency, accountability and trust. (IFRC Report, introduction)

**Globalization**

Policies or strategies for using ICT to deal with the issues of ‘Underserved Communities’ have to start from the concept of information as symbol, and also take cognizance of the context within which the technology is used and developed. The literature on ICTs in the 21\(^{st}\) century is inherently bound up with the onward march of globalization. The term itself is highly contentious, but in this context it needs to be understood to involve free-floating owners and large share-holders having been released from any mutual pact with specific groups of workers or employees, since there is no longer any basis for anchoring a factory, warehouse, depot, or whatever to a particular location. If the costs and factors of production or distribution are more attractive elsewhere – i.e., cheaper, less regulated, non-unionized – then there is little reason to postpone relocation.

This is a key feature of 21\(^{st}\) century globalized production, and in essence it exemplifies the double-edged nature of globalization itself. This has been termed globalication – a flexible and accommodating, global stage for a few, a constrained localized existence for the many. Bauman has also characterized this division as one between tourists and vagabonds. The tourists can choose to travel, and if they do so, they inhabit a world that is both nowhere and anywhere – the airports, the hotels, the restaurants and shops; the technology are designed to be almost entirely independent of any specific location. Indeed, this vacuous universality is often a key feature of the appeal of such amenities, and is extolled in their own marketing. The vagabonds, on the other hand, do not travel from choice, but from necessity. If the employment prospects or life-chances are better elsewhere, then those consigned to a locality may be tempted or impelled to move elsewhere – although increasingly such mass movements are actively discouraged as governments instigate policies to protect their domestic labour force and restrict immigration. Thus the globalized world is one where goods and services can move across borders during their production, distribution, consumption, obsolescence, and extinction\(^{15}\), but where those fundamentally involved in the processes themselves are shackled to their locality – and where the attraction or repulsion of that locality is largely at the behest of forces outside the control of local or even national political forces.

This is a summary of the negative aspects of globalization, and one response to it is to seek refuge in the concept of community; seeing enforced localization as a good thing in itself, from which some new or reconstituted form of

\(^{14}\) The term ICT is preferred to IT, particularly in the context of this paper.

\(^{15}\) The Basel Action Network reports highlight the ways in which poor nations have become the toxic dumps for richer nations’ technological detritus, thus completing the cycle from production to obsolescence – see http://www.ban.org/BANreports/10-24-05/
solidarity may emerge, facilitated by the potentialities of ICTs: Hence a concern with the ways in which ICT might assist ‘Underserved Communities’. Yet the problems with this aspiration arise from the very ubiquity of the term community itself. In the aftermath of just about any major, and all-too-often tragic event, there is near unanimous journalistic recourse to stories about how the ‘local community comes together’; examples are never hard to find. Yet the very fact that such overt and repeated statements have to be made ought to act as a warning that perhaps the term itself is losing its meaning and its reference. This resonates with the observation attributed to Ralph Waldo Emerson – ‘The louder he proclaimed his honour, the faster we counted the spoons.’ With regard to communities in the 20th century, the historian Eric Hobsbawm has observed, ‘never was the word community used more indiscriminately and emptily than in the decades when communities in the sociological sense became hard to find in real life’ (1994, p.428).

In the opening decade of the 21st century it is hard to demur from the argument that flows from Hobsbawm’s observation. The negative forces of globalization have exacerbated the community-destroying forces of the 19th and 20th centuries. Bauman has pointed out that there are two tendencies that have ‘accompanied modern capitalism throughout the whole of its history’, each aimed at creating communities in the aftermath of the massive social upheavals that destroyed the earlier forms of social solidarity. The first was ‘a consistent effort to replace the natural understanding of bygone community, the nature-regulated rhythm of farming and the tradition-regulated routine of the craftsman’s life by an artificially designed and coercively imposed and monitored routine. The second tendency was a much less consistent (and belatedly undertaken) attempt to resuscitate or create ab nihilo a community feeling this time within the framework of the new power structure.’ (2001, p.34, stress in the original). The first tendency was embodied in the awful reality of early factory production, and later resulted in the assembly-line andTaylorism. The second, largely in response to the first, led to efforts to build model villages around and integrated with factories – e.g. Saltaire, Port Sunlight, Bournville in England, and Robert Owen’s projects at New Lanark, Scotland and New Harmony, Indiana, USA. As Bauman notes, these communities did not survive, and the idea that workers might require a more meaningful existence in place of the inhuman routinization of Taylorist factories only re-appeared in the 1930s following Elton Mayo’s work and the emergence of ‘the human relations school’ of industrial sociology. Crucially the success of this latter approach was guaranteed once it became clear that ‘job satisfaction and a friendly atmosphere might go further than strict rule enforcement and ubiquitous surveillance in promoting efficiency at work’ (Bauman 2001, p.37); in other words it turned out to be both more productive and less costly.16

But this is not to allow that such human-centeredness might itself lead to a revival in communities, for Bauman’s argument is that modernity allows no trust in the spontaneous emergence of community; no alternatives can be permitted that might undermine the stability and order ‘designed using the power of reason and maintained by day-to-day monitoring and management’ (p.38). In the globalized context of 21st century modernity this monitoring and management can be done increasingly at a distance, and as was pointed out before, those doing the monitoring need no longer act in loco parentis; they can simply move on, leaving despair, destitution and disintegration in their wake. As Sennett has remarked, domination from the top has become shapeless while losing nothing of its strength. Moreover order can be maintained and authority exerted by the winners, but with a strategy of ‘secession of the successful’.17

All these trends have come together to result in a view of the world that allows no alternative to the command-and-control model. The rich and powerful continue to exert their influence and domination in the sense characterized by Crozier in his analysis of bureaucracy (1964). They maximize their freedom of manoeuvre and opacity, while imposing strict rules and routines on subordinates. Those in control act as the prime sources of uncertainty of those under their control. This is the pessimist’s view of globalization. Yet there is a more optimistic view, one based on a model of cooperation rather than community: the open-source model: An approach that shouldn’t work but appears to do so, albeit in specific, and for the most part virtual, contexts.

16 Within the IS academy the socio-technical movement – owing its origins in part to the influence of those at the Tavistock Institute – appears to be part of this same development. Proponents of the socio-technical movement would probably wish to argue that they constitute a more critical and challenging option; although opinions differ on the effectiveness of such challenges.

17 The phrase was coined by Robert Reich, and is used by Bauman as a chapter heading – see Bauman 2001.
Learning from Open-Source

As was noted earlier, Raymond’s focus is on a collectivity of sorts, but it is one intimately associated with a specific task or project, and it may or may not have some more durable existence beyond those confines. In this regard it may escape some of Bauman’s strictures. In any case the key point for our purposes is that there needs to be a participative orientation from all sides, hence Raymond’s observation of the important roles played by co-developers and beta-testers. He even offers two pertinent observations as specific guidance concerning the ways in which the efforts and enthusiasms of associates or volunteers can best be harnessed and sustained. ‘The next best thing to having good ideas is recognizing good ideas from your users. Sometimes the latter is better.’ [#11] ‘Often, the most striking and innovative solutions come from realizing that your concept of the problem was wrong.’ [#12] And even more insightfully he states that – ‘Given a large enough beta-tester and co-developer base, almost every problem will be characterized quickly and the fix obvious to someone.’ [#8]

What we have here is a call for a truly co-operative venture, whether or not we choose to call it a community. Furthermore it is one owing its continued existence to the widespread availability of ICTs. This raises the issue of the extent to which we might start to develop ICT projects for ‘Underserved Communities’ along the lines suggested by the examples of open-source, and particularly Linux collaborations. Their experiences appear to offer an alternative and provocative blueprint for such projects. Yet anyone reading Raymond’s work at this point might raise what appears to be a major objection, since he states that ‘one cannot code from the ground up in bazaar style’. In other words you cannot build your project from scratch along the lines of a bazaar, something must already be in existence. This is not really a problem, merely recognition of the likely context of any such endeavour, building on what is already there, but doing so in a fashion that mimics the open-source experience rather than the command-and-control bureaucratic one. Furthermore it makes it clear that, for all its faults, the cathedral model has delivered much of the general infrastructure upon which we rely. Raymond stresses that

When you start community-building, what you need to be able to present is a plausible promise. Your program\(^{18}\) doesn't have to work particularly well. It can be crude, buggy, incomplete, and poorly documented. What it must not fail to do is convince potential co-developers that it can be evolved into something really neat in the foreseeable future. (p.47)

The two examples mentioned previously – the phoned warning from someone in Singapore to his family in India, and the people in Tamil Nadu co-ordinating their communications – each built upon existing structures and relationships, together with appropriate use of ICTs. In the former case the radio report was relayed by phone, and then spread around the village in India. In the latter case a network developed based on the locality in Tamil Nadu, incorporating both ICTs and word-of-mouth. What is common to both is that it was not simply communication that mattered, but communication based upon personal relationships and existing networks, using ICTs.

Again Raymond provides an observation that can be readily revised for our purposes. ‘Good programmers know what to write. Great ones know what to rewrite (and reuse).’ [#2] Which can be restated as – Effective activists know what is required. Great activists know what is already in place or available.

Moving Beyond Open-Source – The Agora

Raymond’s distinction between the cathedral and the bazaar is starkly drawn, and has rightly had the effect of drawing attention to the novelty and prime characteristics of the open-source movement. The detail is rather more complex, but that is not in any way a criticism of Raymond’s writing, the power of which comes precisely from his distinctive terminology and accessible style – thus the other chapters of his book include ‘Homesteading the Noosphere’ and ‘The Magic Cauldron’. Yet Raymond’s evocation of the bazaar, with its babble of different agendas, is somewhat misleading; a more extended consideration of the argument will help to clarify some key features of the open-source model, and emphasize the ways in which it might begin to be utilized and incorporated as a community strategy.\(^{19}\)

---

\(^{18}\) Raymond refers here to computer programs, but the point is still relevant if the paragraph is read as referring to community-based programs.

\(^{19}\) Raymond’s response to an earlier draft of this paper makes it clear that he vehemently disagrees with the way in which I have developed the ideas in this section. Raymond is a firm advocate of the free market, and he took exception to my later reference to Margaret Thatcher. He approvingly refers to Ayn Rand in the chapter Homesteading the Noosphere. His political orientation is
Raymond uses the term ‘bazaar’ to evoke the image of a local market place, with many vendors all displaying their wares and competing for custom. To an extent this may have some relevance, although Raymond’s characterization of the open-source mode of operation is distinctly not one of ‘doing business’; there are no monetary transactions, nothing is bought or sold. So perhaps the term bazaar is ill-advised if not ill-chosen. In fact Raymond seems to have used the term because he is following the logic of Adam Smith with his concept of the invisible hand, whereby people pursue their own individual interests but thereby ‘promote that of the society more effectually than [anyone who] really intends to promote it’. In other words, although each and every person is indeed following their own agenda, somehow the end result is not a zero-sum, but a win-win.

The Linux world behaves in many respects like a free market or an ecology, a collection of selfish agents attempting to maximize utility which in the process produces a self-correcting spontaneous order more elaborate and efficient than any amount of central planning could have achieved. (Raymond, 2001, p.52)

But although this is Raymond’s own description of the open-source community, it is not in fact a very accurate summation of how the open-source community operates even as Raymond describes it himself. He seems to see the weakness in his own reasoning and to some extent seeks to remedy it by noting that:

The utility function Linux hackers are maximizing is not classically economic, but is the intangible of their own ego satisfaction and reputation among other hackers. (p.53)

But this raises as many questions as it answers. Who are the other hackers? Does each individual hacker wish to garner the praise of certain big-hitters or simply to amass the admiration of as many other hackers as possible? Are ego satisfaction and reputation assessed by quality or quantity? Does one gain admiration – ‘brownie-points’ – because one achieves some specific result or because one takes large risks? How is success measured? How is success balanced by non-success or downright failure?²⁰

Raymond himself notes that hackers participate as ‘volunteers in an anarchist’s paradise’. The context is co-operative, and not easily subsumed under some risk-reward calculus or utility function. The term bazaar in this context is misleading; there is a far more appropriate one which Raymond himself uses in a final remark he appended to a later version of the original paper.

Finally, I must admit that I very nearly called this paper The Cathedral and the Agora, the latter term being the Greek for an open market or public meeting place. The seminal agoric systems papers by Mark Miller and Eric Drexler, by describing the emergent properties of market-like computational ecologies, helped prepare me to think clearly about analogous phenomena in the free-software culture when Linux rubbed my nose in them five years later. (pp.225-6)

Miller and Drexler’s characterization of the agora sees it as inherently market-oriented and exchange-oriented. This is a common, but constrained and restricted, view.

This line of investigation leads us to propose what may be called the agoric approach to software systems. Agoric stems from agora, the Greek term for a meeting and market place. An agoric system is defined as a software system using market mechanisms, based on foundations that provide for the encapsulation and communication of information, access, and resources among objects. Each of these notions plays a role in supporting computational markets. (stress in original - http://www.agorics.com/Library/agoricpapers/aos/aos.1.html)

Miller and Drexler’s key concern is to outline an argument for applying a model oriented around ‘decentralized market co-ordination’ to the development and sustainability of complex computational systems. Raymond clearly feels that the open-source context is akin to this, hence his reference to ‘agoric systems’. But he also offers a glimpse of something different and distinctive: In fact something far more like the original meaning of the term agora.

very clear, so it is not surprising that he scornfully replied to an article accusing him of purveying a ‘socialist … or vulgar Marxist … interpretation of software development’. Nothing could be wider of the mark. http://www.catb.org/~esr/writings/response-to-bezroukov.html

²⁰ Raymond does deal with these issues in the later chapters of his book – but the general points made here still stand. The economic analysis of open-source is a developing topic of discussion and an important one – see Weber, 2005 and Raymond, 2001.
The *agora* in Ancient Greece was a specific location. Initially it was the place for public assemblies, and only later was it also used as a market place. The *agora* as a concept, however, is critically different from the concept of the market. The market is a space of exchange, where everything has to be available as an exchange-value, and agents can buy and sell commodities at whatever is deemed to be the going rate. By definition and design the market is mechanistic – hence the term market mechanisms; and it is non-human in the sense that it does not operate on the basis of people’s beliefs and priorities unless they can be translated into prices and exchanges. In contrast the original concept of the *agora* was distinctively human and collective, the space between the private realm, the *oikos*, and the public realm of the state, the *ecclesia*. Bauman defines it as

the space neither private nor public, but more exactly private and public at the same time. The space where private problems meet in a meaningful way – that is, not just to draw narcissistic pleasures or in search of some therapy through public display, but to seek collectively managed levers powerful enough to lift individuals from their privately suffered misery; the space where such ideas may be born and take shape as the ‘public good’, the ‘just society’ or ‘shared values’. (1999, pp.3-4)

For Bauman, the history of modern societies has been a long war of attrition ‘launched against the agora from the side of the ecclesia’. In other words the state has sought to curtail or eradicate this space in which issues pertaining to the collective, the shared, the communal, can be raised and discussed. In the 1980s this goal of curtailment found its expression in the infamous statement of Margaret Thatcher, at that time the British Prime Minister, that ‘there is no such thing as society…there are individual men and women, and there are families’. In fact the sweeping changes associated with Thatcherism, such as privatization and encouragement of market freedoms, intensified the assault. At the same time as the state sought to undermine any claims for legitimacy for the social and communal, it was also enacting policies that resulted in the expansion of the operation and calculus of the market flooding into all aspects of human existence – social and personal, the public and the private. The novelty of this was in its intensity rather than its actual occurrence. As Hannah Arendt (1998) pointed out, the pressure from the *ecclesia* often took the form of efforts to transform the *agora* ‘into an assemblage of shops like the bazaars of oriental despotism’ (stress added).

The open-source community is perhaps an example of a way in which this process might be reversed, a glimpse of the *agora* in its original sense, albeit one that goes against the grain of Raymond’s wider argument. The Linux community appears to exemplify the emergence of a group whose exchanges are not predominantly market-based, but oriented by something more akin to ‘the public good’, although the extent and nature of the ‘public’ who are supposed to benefit from this ‘good’ is a somewhat limited one. Again it must be noted that Raymond emphatically does not see it in these terms, since, as Hannah Arendt (1998) pointed out, the pressure from the *ecclesia* often took the form of efforts to transform the *agora* ‘into an assemblage of shops like the bazaars of oriental despotism’ (stress added).

The open-source community is perhaps an example of a way in which this process might be reversed, a glimpse of the *agora* in its original sense, albeit one that goes against the grain of Raymond’s wider argument. The Linux community appears to exemplify the emergence of a group whose exchanges are not predominantly market-based, but oriented by something more akin to ‘the public good’, although the extent and nature of the ‘public’ who are supposed to benefit from this ‘good’ is a somewhat limited one. Again it must be noted that Raymond emphatically does not see it in these terms, since, as Hannah Arendt (1998) pointed out, the pressure from the *ecclesia* often took the form of efforts to transform the *agora* ‘into an assemblage of shops like the bazaars of oriental despotism’ (stress added).

The *Wiki Movement*

The term *Wiki* seems to have several meanings and derivations. The word itself means ‘quick’ or ‘fast’ in Hawaiian, and the slogan *WikiWiki* is used by a shuttle bus company at Honolulu International Airport. It is also claimed that Wiki is an acronym for ‘What I Know Is’. Hence the term has come to denote collaborative efforts where people come together to pool their knowledge and expertise with a minimum of fuss and formality. There is a large and growing literature on Wiki principles and philosophy, naturally with various contending camps and positions emerging. In many regards the Wiki principles are more easily understood from stating what the Wiki movement is not, rather than what the Wiki movement actually is. Hence the following headings from the Wikipedia entry on Wikipedia itself:

What Wikipedia is not
1.1 Wikipedia is not a paper encyclopedia
1.2 Wikipedia is not a dictionary
1.3 Wikipedia is not a publisher of original thought
1.4 Wikipedia is not a soapbox
1.5 Wikipedia is not a mirror or a repository of links, images, or media files
1.6 Wikipedia is not a free host, blog, or Web space provider
1.7 Wikipedia is not an indiscriminate collection of information
1.8 Wikipedia is not a crystal ball
1.9 Wikipedia is not censored for the protection of minors

What the Wikipedia community is not
2.1 Wikipedia is not a battleground
2.2 Wikipedia is not an experiment in anarchy
2.3 Wikipedia is not a democracy
2.4 Wikipedia is not a bureaucracy

In fact what the organizational model relies on is a sufficient number of people feeling motivated and enthused to contribute and participate, exactly the same prerequisites that Drucker identifies for volunteers or associates, and that Raymond describes for the Linux participants. Moreover what the Wiki philosophy has in common with both open-source development, and with Brooks, is that such endeavours are best seen as cultivation as opposed to construction. Brooks mentions the move from seeing software as being built to seeing it as grown. Similarly, astute readers might have noted that in the extract used earlier, Raymond sees the Linux world behaving ‘like a free market or an ecology’ (stress added). In all these cases there is an absence of overall or grand design, but an emergent development based on constrained and localized initiatives. So what they share is that at any one time they are less organized than perhaps they might be, but they are also more extensive, more accessible, more visible, and more speedily updated and corrected than standard command-and-control centralized systems.

It is true that the Wikipedia protocols have had to be strengthened in the light of various operational problems, and Raymond in the other chapters of his book discusses the forms of conflict-resolution used by the open-source movement, but in general the key ideas about their operation remain unaffected. The reality of open-source and Wiki models is that there are forms of co-ordination and control, but they differ from those associated with command-and-control management. Moreover Raymond argues that the overheads required for traditional forms of management cannot be justified, particularly since they do not even deliver what they are meant to do.

**Conclusion**

The open-source experience offers an example of a new way of organizing, one that throws a glance of dim recognition at the agora. In Raymond’s terms some of the key ideas derived from the open-source experience are ‘really neat’ and can form the basis of some tangible and appreciable way of contributing to efforts to benefit ‘Underserved Communities’. There are, however, a number of caveats. The first is that although the open-source and Wiki movements produce a significant, albeit intangible, product, using such models as an organizing principle amongst Underserved Communities – however defined and identified – is qualitatively different. The very process and extent of the involvement of the target group is critical, not simply the orientation of those willing to participate. For open-source and Wiki groups it may well be the case that, above some critical mass, ‘the fewer the better’, but this cannot be true for community-based projects where inclusion is a key objective.

A second point of note is that whereas much discussion has ensued in the context of open-source with regard to ‘free-loaders’ and others who seek to benefit from bazaar-like projects without themselves adding anything to the overall effort, community-oriented projects have to take a different stance. This is why the concept of the agora is important, as it emanates from a view of inclusiveness, assuming that everyone is part of the process of deliberating about and developing the ‘common good’. This may seem to stretch Raymond’s discussion well beyond the bounds he himself set. Yet what has been done is to take the issues raised by his model, and treating them as tools and resources that might be applied to contexts and concerns significantly different from their original source. This is a key principle of the open-source movement, since they built upon tools and expertise that were already in existence. Moreover it is in line with Raymond’s observation that the bazaar model is not a basis for development from scratch. Several of his maxims reinforce this

---

22 A key difference, however, is that the contributors to Wikipedia are for the most part anonymous; this, and the related issue of ‘ego-based motivation’ referred to by Raymond, need further examination, but at another juncture.
point, focusing on re-use of what is already available, reliance on others (e.g. beta-testers), and recognizing good ideas from elsewhere. So the argument presented here is very much in line with these sentiments – again Raymond summarizes this orientation with a pithy aphorism: ‘Any tool should be useful in the expected way, but a truly great tool lends itself to uses you never expected.’[#14] For software development it is assumed that there is a tool-box of tried-and-tested devices ready at hand. Raymond is making the point that great tools should be both accessible and flexible. People can add to the tool-box as well as using and adapting what is already there. One of the criteria for judging the success of ICIS 2006 might be the extent to which a start is made to develop such a resource for Underserved Communities, and the ways in which such tools are utilized and also become used and enhanced in unexpected ways.

Land’s challenge was to identify ‘the work in our community which yields new ideas on how IS can serve mankind’; it may well be that the best current candidate for this is not a particular technology or product, but a mode of organization and development – something for which the IS community is not often held up as a paragon, but something that is ‘really neat’.

Acknowledgements

I would like to thank Frank Land for his helpful and perceptive comments on an earlier draft of this paper. I must also acknowledge Eric Raymond’s contribution, both in terms of his writings on the open-source movement as a whole, and his comments on a very early draft of this paper. As has already been noted, he does not agree with the ways in which I have used and expanded upon his ideas, but thankfully the IS-world operates more like a bazaar than a cathedral. The two anonymous ICIS reviewers offered constructive and supportive ideas for ways of improving this paper for the conference itself, and also for developing my ideas further – a challenge which I hope to take up in the future.

References

Bauman, Z. In Search of Politics, Polity, Oxford, 1999
Drucker, P. “Management’s new paradigms”, Forbes Magazine, 10.05.1998, pp.152-176
Land, F – posting to ISWORLD, 26 March 2004
Raymond, E. “The Cathedral and The Bazaar”, first given at the Linux Kongress, 1997; subsequently republished and revised on several occasions and in several places; the version referenced here is the title essay in Raymond’s book, The Cathedral and The Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary, O’Reilly, 2001 (page references are for the latter edition)