distinction between the 'media person' and the 'ordinary person'. If alternative media as a social practice aims to erode this distinction, then Lievrouw's book becomes an important step towards a study of alternative media rituals as social practice.

References

Bailey, O. *et al*. (2008) *Understanding Alternative Media*, McGraw Hill, New York.
Couldry, N. (2003) *Media Rituals: A Critical Approach*, Routledge, London.
Harold, C. (2007) *Our Space. Resisting Corporate Control of our Culture*, University of Minnesota Press, Minneapolis, MN.
Rodriguez, C. (2011) *Citizens' Media against Armed Conflict*, University of Minnesota Press, Minneapolis, MN.

Chris Atton
Edinburgh Napier University
c.atton@napier.ac.uk

http://dx.doi.org/10.1080/08109028.2012.750442

Internet success: a study of open-source software commons, by Charles M. Schweik and Robert C. English, Cambridge, MA, MIT Press, 2012, 344 pp., £24.99, ISBN: 978-0262017251

In a historical sense, open source software (OSS) development is a relatively new activity that generates interest from software developers and students of innovation alike. The text by Charles M. Schweik and Robert C. English, titled *Internet Success: A Study of Open-Source Software Commons* and published by MIT Press, is therefore timely because it seeks to bring understanding developed through rigorous research to a topic dominated by, as one reviewer puts it, 'hacker folklore' (book sleeve). The reason why the authors have chosen to focus on success is perhaps obvious to those seeking to better manage OSS projects. However, the emphasis on success will act as a dog whistle to those who see failure as perhaps just as instructive to the understanding of OSS.

Indeed, Schweik and English discover that neat distinctions between success and failure are resistant to definition as OSS emerges as a multifaceted concept (pp.129–42). Hence, they qualify their descriptions of success by developing six categories that distinguish projects that initially achieve one official release of software from those that achieve releases of three or more versions. The term 'failure' is judged to be inappropriate and is replaced by the term 'abandonment', which is divided into two sub-categories to reflect abandonment before one release and before three releases, respectively. They also find the need to define situations in which the assessment of success or abandonment is premature or unreliable. These make up the final two categories.

The authors set out to provide a rigorous and empirical assessment of OSS development throughout the world. The borderless nature of the Internet facilitates such an approach and given the masses of information that are available, statistical analysis of site metrics and surveys are used along with interviews with authorita-

tive actors to address over 40 research questions and hypotheses. It is clear that a primary purpose of the book is to provide an account of the research process as much as of the research findings because the book reads very much like a research thesis. The methods used resonate with quantitative methods found in Computer Science and Software Engineering so one presumes that the book will find favour within these quarters. Certainly, this approach is given endorsement by the reviewers on the book sleeve who praise the authors not only for the study's timeliness but also for making their evidence publicly available for others to corroborate the authors' findings.

The question as to whether this is the best format for the presentation of such research is no doubt one the editors struggled with. The research domain is a socio-technical one where complexity leads many researchers in this area to be more circumspect. Indeed, such researchers tend to use a different model in reporting on research findings, particularly in hardcover format. This model allows authors to describe the breadth of their research project in a longish introduction, then focus on different themes in subsequent chapters, thereby building an understanding of the complexities. The decision to keep a structure that obviously follows a classic research thesis is one that didn't find favour with this reviewer.

Perhaps as a consequence of this decision, the book lacks a thesis that enables the pieces of evidence to be connected within the reader's mind. The authors seem to outsource this aspect of their research to the statistical analysis at the end, which ultimately produces some notable correlations but falls short in providing a meaningful understanding of OSS. Dependent variables multiply as each page is turned. At one point, the authors even ask whether programming languages (C++, Perl, PHP and so on) are a factor in OSS success. Hats off to them for leaving no stone unturned. Given their familiarity with the topic, I would have liked their assessment of OSS in a sentence or two.

The authors accept that OSS is just as much a social phenomenon as a technical one. This is demonstrated in Chapter 2 where an overview of OSS projects from an institutional perspective is provided. They discuss the motivations for the adoption of OSS by the private sector, government, not-for-profit organisations and academia. The reader discovers that some motivations for adoption are common to all. For example, all find the need to reduce costs for non-differentiating IT – information technologies that are commonplace in organisations (such as databases, word processing). There are also variations in the motivations to use OSS. Private sector motivation for the use of OSS can be seen in the need to develop services by which companies can be differentiated from their competitors. Government adoption of OSS often stems from the need to promote greater interoperability and information sharing, not only between government departments but also with the citizenry. Not-for-profit organisations adopt OSS because of cost and the greater availability of volunteer labour – OSS aligns well with the open and collaborative culture the authors associate with not-for-profit organisations. Academia has also been enthusiastic in its embrace of OSS for a number of reasons. Once again, cost is a factor but it seems that there is greater pressure on universities to adopt OSS as part of publicly funded research into software development or as part of the requirements of funding agreements. Finally, the authors identify a new not-for-profit institution designed to promote a public face of the OSS movement. Organisations such as Free Software Foundation (FSF) and the Open Source Initiative (OSI) have taken on the role of speaking for OSS in general. Other organisations have been formed

to provide a public face at the project level: Software in the Public Interest, Apache Software Foundation, Perl Foundation and Free BSD Foundation. These organisations play an important role in managing OSS activities with corporations and governments.

Schweik and English investigate the role of institutions in OSS by describing their relationships as an ecosystem but never really define what an ecosystem actually is. They describe an institutional and development framework (IAD) to depict the complex interactions that occur in relation to OSS (pp.38–44). Technological attributes, community attributes and institutional attributes are considered as influences that contribute to unique patterns of interactions, which over time reshape these three main inputs in an iterative fashion. This is where the development of ideas gets confusing because one is initially led to believe that the structure and interactions within institutions are key to understanding OSS. Then they announce that the software developer will also be included as a key unit of analysis (p.44).

So, the authors also consider the incentives that motivate software developers to contribute their time freely as another factor that should be included in the analysis (pp.44–55). While the portrayal of a generous and community-minded software developer is useful for understanding aspects of OSS, the fact emerges that there is money to be made from working on OSS projects. They reveal the different motivations that a budding university graduate in software development may have from the older, more experienced programmers at the top of their game. The former is more willing to give time freely in order to develop experience and status within the community while the latter are less motivated to give their time freely with the possible exception of those who wish to enhance their leadership credentials within a developer community – something that the authors describe in terms of meritocracies (p.47). Other attributes that emerge as important are the knowledge and experience developers have gained in specific projects. Some view OSS as a social movement of the kind championed by Richard Stallman in which software is viewed as a public good (p.52). The hacker ethic and reciprocity are values that are seen to define membership of the OSS community. These values seem to appeal to males as the authors found that 98% of the developers who responded to their survey were male (p.309). While it is no secret that males dominate Computer Science and Software Engineering courses, this figure is higher than the 88% proportion of Computer Science graduates in the United States who are male (Zweben, 2012).

The theoretical framework for the study therefore seeks to include the various influences that technology, community attributes, institutional design and software developers have on OSS projects. There are over 40 hypotheses and research questions as listed below – a number that the authors admit is difficult to adequately address (p.297). There is not sufficient space here to deal with these points individually but they are covered extensively in the book. Many of the 351 pages of the book are devoted to providing such detail.

- OSS commons community attributes (user involvement)
- Technological attributes
  - Software requirements
  - Software design: modularity, granularity and complexity
  - Product utility

- Competition
- Collaborative infrastructure

- Community attributes of OSS commons
  - User involvement
  - Leadership
  - Social capital
  - Group heterogeneity
  - Group size
  - Financing
  - Marketing

- Institutional design
  - Understanding OSS institutions
  - Investigating the role institutions play in project success and abandonment

The authors were able to draw on two significant cases for their research: OSGeo and Source Forge. The first case was used to initially validate the concepts proposed in their research questions and hypotheses (pp.103–28). OSGeo is an umbrella organisation designed to support and promote the development and use of non-proprietary open source mapping software or, as it is more commonly referred to, geographical information systems (GIS). There is a strong OSS presence in this domain, which has many links to both government and private enterprises throughout the world, including developing countries. The authors use the OSGeo case to demonstrate the complexity of interactions that they claim characterises OSS projects more generally.

The second case of Source Forge provides the data for statistical analysis. Source Forge is the largest OSS project-hosting website in the world and provides access to 2.7 million developers and 260,000 OSS projects (p.130). Source Forge is analysed from two perspectives. The first is to look at a selection of projects and explore key metrics of these projects such as numbers of releases, abandonment, number of downloads, tracker reports and so on. The outcome of the analysis is contained in the supported hypotheses that have been linked with success (one release or three releases of software). The supported hypotheses relate to: a clear vision; the utility of software to end-users; and leadership skills to coordinate and inspire team members (p.169). This is further summarised in the following statement:

> In short, projects with members who have leadership skills that build community along with management skills that make good use of collaborative infrastructure will be more likely to succeed in the growth stage. (p.173)

It is reassuring to discover that a statistical analysis of the numerical data supports what some would consider as banal.

The authors remain with Source Forge to conduct further research that draws on the knowledge of project administrators using quantitative survey methodology. From a potential sample size of 49,000 individuals, the invitations to participate in the survey yielded 1403 usable respondents (p.187). Readers are taken through the many steps Schweik and English took to ensure that a reliable sample had been

achieved. Even though the response rate appears low the authors were satisfied that the sample was representative of OSS projects on Source Forge.

The authors experienced many challenges. For example, questions about social capital were skewed by the fact that over half of the projects reviewed had only one developer (pp.234–39). Hence, questions about the frequency of meetings with colleagues were not relevant. The authors explain that this unforeseen bias in data structure skewed the answers of those who were able to answer this question. In any case, the authors go on to find that social capital, as defined by face-to-face meetings, was not found to be a significant factor contributing to the perceived success of OSS projects.

The final chapter of the book brings together the findings of the research. The authors first give advice to those who are venturing into an OSS project by outlining the 'practical implications' of their study (pp.303–4). They summarise initiatives that were found to be associated with successful OSS projects. These include hard work, good leadership, well-articulated vision, high quality website, marketing and creating software that many people will find useful. Participation in GPL (GNU Public License) projects was not a factor indicating future success.

For their academic audience, they go on to summarise each of the theoretical issues that were the source of their research questions and hypothesis. The authors only devote a few paragraphs to each of the following points.

- Motivations for participating in OSS
- OSS commons and collective action theory
- OSS, face-to-face meetings and social capital
- OSS and group size
- OSS institutions

As stated earlier, the absence of a unifying thesis that these issues address is a disappointment. The closest that the authors come to articulating a central thesis is when they devote a few pages to discussing the relevance of their OSS research to the rise of global online social movements aimed at addressing problems of social significance such as climate change. Once again they present their reflections within the context of the previously outlined theoretical insights but do not take the step of 'joining the dots' about where OSS fits into the world as a human activity.

The obligatory section on limitations at the end provides further insights about the authors and their assumptions about research. They warn readers that self-reporting behaviour of respondents cannot be verified. This limitation should have been recognised from the outset and addressed, perhaps, by using multiple research methods. In my view, the greatest limitation of the research is the authors' ignorance of the broader landscape of innovation research.

So readers expecting to find a fundamental rationale from which all of these differing attributes of the OSS movement stem will be disappointed. Despite this apparent shortcoming, the book defines a number of worthwhile research questions that can be taken up by student researchers. In fact, those with a penchant for statistics and who may wish to replicate their study or extend it further are able to make use of their data, which have been posted online for others to use.

Reference

Zweben, S. (2012) *Computing Degree and Enrollment Trends: From the 2010–2011 CRA Taulbee Survey*, Computing Research Association, Washington, DC.

William Tibben
School of Information Systems and Technology, University of Wollongong, Australia
wjt@uow.edu.au

http://dx.doi.org/10.1080/08109028.2013.767423

'Ribbon of Fire' How Europe adopted and developed US strip mill technology (1920-2000), edited by Jonathan Aylen and Ruggero Ranieri, Maesteg, Mid Glamorgan, Pendragon/Fondazione Ranieri di Sorbello, 2012, 410 pp., €45, ISBN 978-8865982389

This book is the belated result of a conference of experts held in Manchester in 2001. Although the experts apparently produced texts on their individual topics, the editors and others have put another 10 years of work into extending and refining the contributions. The result is an academic text akin to a dissertation presented for the attainment of a higher university degree. It really should come with a health warning: "not suitable for readers without thorough knowledge of the steel industry". The book contains a glossary of terms, but again, this is not sufficient to enlighten the uninitiated. The text was published in digital form and, as an afterthought, in book form.

The book consists of three parts, each authored by a different expert:

Part I Wide strip mills: economic and technical; developments

Part II Case studies in wide strip mill installation and development

Part III Plant suppliers, automation and users: the wider context of wide strip mill development.

Part I looks like the most enlightening part of the book, but is very hard to understand without prior detailed knowledge of steel rolling procedures for the production of various forms of sheets or strips of steel. Steel in these forms is in great demand for the production of automobile bodies, cans for food and drink products, and a variety of white goods, such as refrigerators. The wide strip mill (WSM) is an American invention which underwent several stages of further development. Most of these stages happened between about 1920 and 1940. Some thinning of the steel was achieved by cold rolling, often as a further stage after WSM. The advantage of cold rolling is that it produces a better surface finish and ductile steel suitable for deep drawing, making it ideal for automobile body manufacture. At the end of the shaping and thinning processes, the strips are usually formed into large rolls and sold in this form. As an integral part of the manufacture of strips or sheets of mild steel, various coating procedures can be applied. The steel is often coated with zinc either by electrolysis or by hot dipping to protect it from corrosion, or with tin to make it suitable for the production of tin cans.

The introduction of WSM was a true revolution in steel making. It reduced the cost significantly and increased productivity. The authors make the point that many