This article was downloaded by: [216.73.216.198] On: 22 June 2026, At: 13:46
Publisher: Institute for Operations Research and the Management Sciences (INFORMS)
INFORMS is located in Maryland, USA

Management Science

Publication details, including instructions for authors and subscription information:
http://pubsonline.informs.org

Management Insights

To cite this article:
(2006) Management Insights. Management Science 52(7):iv-vi. https://doi.org/10.1287/mnsc.1060.0590

Full terms and conditions of use: https://pubsonline.informs.org/Publications/Librarians-Portal/PubsOnLine-Terms-and-Conditions

This article may be used only for the purposes of research, teaching, and/or private study. Commercial use or systematic downloading (by robots or other automatic processes) is prohibited without explicit Publisher approval, unless otherwise noted. For more information, contact permissions@informs.org.

The Publisher does not warrant or guarantee the article's accuracy, completeness, merchantability, fitness for a particular purpose, or non-infringement. Descriptions of, or references to, products or publications, or inclusion of an advertisement in this article, neither constitutes nor implies a guarantee, endorsement, or support of claims made of that product, publication, or service.

Copyright © 2006, INFORMS

Please scroll down for article—it is on subsequent pages

With 12,500 members from nearly 90 countries, INFORMS is the largest international association of operations research (O.R.) and analytics professionals and students. INFORMS provides unique networking and learning opportunities for individual professionals, and organizations of all types and sizes, to better understand and use O.R. and analytics tools and methods to transform strategic visions and achieve better outcomes.
For more information on INFORMS, its publications, membership, or meetings visit http://www.informs.org

MANAGEMENT SCIENCE
Vol. 52, No. 7, July 2006, pp. iv–vi
ISSN 0025-1909 | EISSN 1526-5501 | 06 | 5207 | 00iv

informs

DOI 10.1287/mnsc.1060.0590
© 2006 INFORMS

Management Insights

Downloaded from informs.org by [216.73.216.198] on 22 June 2026, at 13:46 . For personal use only, all rights reserved.

The Promise of Research on Open Source Software
Georg von Krogh, Eric von Hippel

Open source software projects break with many established assumptions about how innovation *ought* to work. As a result, research focused on these projects can offer wonderful examples of novel innovation practices and major new insights to students and practitioners in many fields. For example, open source software projects involve networks of independent software developers working with minimal coordination. Yet complex, effective software products like Linux are routinely produced by these projects. In this article we briefly review existing research on the open source phenomenon and discuss the utility of open source software research findings for many other fields. We categorize the research into three areas: motivations of open source software contributors; governance, organization, and the process of innovation in open source software projects; and competitive dynamics enforced by open source software. We introduce the articles in this special issue of *Management Science* on open source software, and show how each contributes insights to one or more of these areas.

Understanding the Motivations, Participation, and Performance of Open Source Software Developers: A Longitudinal Study of the Apache Projects
Jeffrey A. Roberts, Il-Horn Hann, Sandra A. Slaughter

Open source software (OSS) plays an increasingly important role in IT infrastructure and product development decisions, but has been viewed critically and lacks wide-spread adoption. An often cited concern is the dependability of volunteers whose motivations are not well understood. Conversely, increasing commercial OSS participation raises questions of whether extrinsic incentives, such as pay, may diminish the intrinsic motivations of OSS contributors. In this paper we offer the following insights into how an OSS community attracts and sustains participation. (1) Paid participants have significantly above average levels of contributions. Further, extrinsic incentives of contributors do not compromise their intrinsic enjoyment of OSS participation. This bodes well for many successful OSS projects experiencing increased levels of commercial involvement. (2) Firms can rely on rewards that appeal to "ego" to influence OSS participation and performance. The status and career concern motivations of OSS contributors significantly influence their levels of participation. In addition, these motives are positively influenced by extrinsic incentives. OSS communities may want to nurture such egocentric motivations via public recognition of distinguished contributors or by promoting participation as leverage in the labor market. (3) OSS communities benefit from a formal feedback system that rewards personal competence. Increased recognition is associated with increased status motives, which are associated with increased participation.

Motivation, Governance, and the Viability of Hybrid Forms in Open Source Software Development
Sonali K. Shah

We see increasing evidence of innovative activity occurring outside the boundaries of firms and research institutions. Users, often working within the social structure provided by *innovation communities*, are influential in the creation and diffusion of important innovations in fields as diverse as automobiles, sports equipment, personal computers, and software. As managers become increasingly aware of the existence and relevance of innovation communities, they must decide how they will work with them. In this paper we investigate how sponsorship by firms can affect the motivations and contributions of voluntary participants in software development communities. Specifically, we examine the impact of two broad sets of governance mechanisms—decision rights and property rights—on the actions of volunteer developers. When these rights are held by a firm, rather than by the community, voluntary developers contribute less to the community, possess distinctly different motivations for contributing, and take on fewer code maintenance tasks. As a result, the use of governance mechanisms that directly permit *value appropriation* by the firm can be detrimental to *value creation* within the community. Firms seeking to construct hybrid arrangements that balance community-based value creation with private value appropriation must find or establish a community with similar end goals, allow authority within the community for setting project direction and owning project outputs, and meet the community's standards of fairness and reciprocity.

Exploring the Structure of Complex Software Designs: An Empirical Study of Open Source and Proprietary Code
Alan MacCormack, John Rusnak, Carliss Y. Baldwin

Two products intended to perform the same function often have quite different designs. What drives these

Downloaded from informs.org by [216.73.216.198] on 22 June 2026, at 13:46 . For personal use only, all rights reserved.

different choices, and what are the implications? We have developed a set of tools and metrics to help answer such questions in the software industry. These allow us to visualize the structure of software designs and measure their degree of modularity in comparison to other designs. In this paper, we use these methods to compare a product developed via "open source" methods to a product developed by a commercial software firm. We find that the open source product is significantly more modular, mirroring the distributed nature of its development team. But we also find that a purposeful effort to redesign the commercial product resulted in a design of comparable modularity. Our findings suggest that recent moves to release commercial software into the public domain may fail unless the product possesses an architecture for "participation." More broadly, our work highlights that a product's design is determined not only by the functions that it performs, but also by the choices that designers make as they respond to the incentives and organizational structures that surround them. Managers must recognize the impact of these factors to better understand the design choices available to them.

Strategic Interaction and Knowledge Sharing in the KDE Developer Mailing List
George Kuk

This paper examines the dynamics of knowledge sharing in the K Desktop Environment (KDE) developer mailing list. Specifically, we investigate why much of the development work is done by a small subset of developers despite the availability of tens of thousands of developers worldwide. The investigation leads to a number of insights for managers practicing and promoting knowledge-sharing initiatives using developer mailing lists. (1) Managers should not be alarmed and hastily withdraw support when there is a moderate concentration of knowledge sharing. (2) Managers should recognize that highly resourceful developers often behave strategically in the way they select which discussion forums to join and who to engage in social exchanges. Hence, managers should avoid recruiting codevelopers on their behalf. (3) Managers can use the measures of this study to gauge knowledge sharing within and across discussion threads. These can be used to guide future decisions regarding when to intervene and what actions to take to drive and sustain knowledge sharing and integration. Specifically, the measures can be used to differentiate situations where only a few developers monopolize the discussions from the ones where concentration helps intensify interactions and knowledge sharing.

Location, Location, Location: How Network Embeddedness Affects Project Success in Open Source Systems
Rajdeep Grewal, Gary L. Lilien, Girish Mallapragada

Individuals participating in online communities to develop open source software are connected to each other through the projects that they participate in; thereby creating a network of relationships constituting both projects and developers. In this research, we investigate how the network structure that evolves around open source projects can explain differences in technical sophistication and software adoption across projects. Our results show that having a more central project location has a greater influence on technical success than on commercial success and projects with more developers experience greater technical success as they age. We also find that the technical success of open source software development projects is jeopardized when the project leader works on a very large number of projects (i.e., is more centrally located in the network). In addition, project location appears more critical in determining technical success than the location of the project manager. These findings suggest that firms (such as IBM and Sun Microsystems) using an open source environment for software development would be better off participating in projects with a larger number of networked developers (i.e., people working on other popular projects), if gaining technical sophistication is their key objective. In contrast, if managers are seeking commercial success in the open source world (i.e., more downloads) then they may be able to overlook the social network of the project and the developers working on it.

Two-Sided Competition of Proprietary vs. Open Source Technology Platforms and the Implications for the Software Industry
Nicholas Economides, Evangelos Katsamakas

We analyze and compare the strategic differences between open source (such as Linux) and proprietary technology platforms (such as Microsoft Windows). We argue that a strength of a proprietary platform comes from the ability to use pricing strategically to influence and coordinate markets for complementary applications that are compatible with the platform. For example, Microsoft effectively subsidizes application developers by providing them with development information and resources, while the Sony game platform collects a fee from application developers. However, developers of proprietary applications compatible with an open source platform are likely to be more profitable than application developers for a proprietary platform because the price of the open source platform is zero and users value systems made up of the platform and applications as single entities. As a result, application developers for an open source platform (e.g., IBM) have incentive to subsidize user adoption of this platform. Total industry profits are likely to be highest when the platform is proprietary, especially when the platform is vertically integrated with applications (because vertical integration leads to better coordination of pricing that maximizes the total profit), or when users value application variety strongly (because the open source platform is unable

Downloaded from informs.org by [216.73.216.198] on 22 June 2026, at 13:46 . For personal use only, all rights reserved.

to capture a profit out of this value). Finally, when proprietary and open source systems compete, the proprietary system is likely to dominate the open source system in terms of market share and profits, provided that there are more applications compatible with the proprietary platform.

Dynamic Mixed Duopoly: A Model Motivated by Linux vs. Windows
Ramon Casadesus-Masanell, Pankaj Ghemawat

We analyze a simple model that captures aspects of the competition between Linux (open source software, or OSS) and Microsoft's Windows (for-profit proprietary software). Our analysis allows for strategic behavior by Microsoft—i.e., recognition of how its own choices will influence the evolution of Linux—and contradicts the common assertion that given faster demand-side learning, OSS development efforts will overtake or even oust proprietary development effort from markets. Other common assertions about OSS that our analysis contradicts or qualifies include the ideas that OSS is inherently incompatible with the protection of property rights, that governmental commitments to promote OSS ensure lower prices, and that forward-looking buyers always tip market outcomes toward OSS and away from proprietary software.

Entry Strategies Under Competing Standards: Hybrid Business Models in the Open Source Software Industry
Andrea Bonaccorsi, Silvia Giannangeli, Cristina Rossi

Open source software has changed the way in which a key element of the information society is produced and distributed. Considerable discussion and conflict has followed this innovation. Based on a survey of software firms producing open source solutions, we found that few have adopted a business model based on extreme alternatives, either full proprietary or pure OS. Instead, firms typically adopt a hybrid business model, by using OSS for some markets and proprietary solutions for others, and combining different types of OSS licenses. Specifically, we find that in markets dominated by proprietary standards and subject to strong network externality effects, firms accept proprietary products, whereas in markets in which OS solutions have reached or are close to reach the critical mass of adoption, firms push these solutions with their customers. The dominance of hybrid business models means that firms wondering about whether and how to adopt OSS can take a gradual approach, by testing solutions against proprietary ones on a case by case base. In addressing OSS firms they can expect to receive pragmatic and supportive advice, rather than fundamentalist guidance.

Open Source Software User Communities: A Study of Participation in Linux User Groups
Richard P. Bagozzi, Utpal M. Dholakia

The success of open source software (OSS) projects hinges on active (and volunteer) participation in user groups, which perform a variety of crucial marketing, service support, and business-development functions at the grassroots level. In this paper, we report the results of a large-scale survey conducted to examine why users participate in and contribute to OSS user groups. Using a structural equation modeling methodology, our research reveals that participation in OSS user groups can be explained by a combination of social and psychological variables. We also study the issue of whether user participants become more or less engaged in OSS user groups as their experience level increases. Our results indicate that members are more influenced by the group with greater experience, and tend to engage in significantly more joint social interactions, when compared to novice members. Taken together, these findings suggest that OSS user groups are important conduits for fostering loyalty and engagement with the software and the project among new and existing OSS users. Project organizers are advised to devote significant attention to organizing and facilitating networks of user groups to take advantage of these effects. All new software users should be actively encouraged to join a user group and participate in it on adopting the product.

The Architecture of Participation: Does Code Architecture Mitigate Free Riding in the Open Source Development Model?
Carliss Y. Baldwin, Kim B. Clark

In the past ten years, open source software development has become a central driver of change in the worldwide computer industry. To make effective decisions concerning their firms' relationships to open source codebases, managers need to make informed judgments about a given code's long-term viability and quality. In this paper we explore the relationship between the architecture of the codebase and open source developers' incentives to contribute to its development. We argue that a modular codebase and/or one with high inherent option value is likely to attract more voluntary effort, achieve higher levels of quality, and exert more discipline on competitors than one that is not modular or has little inherent option value. Hence, when formulating software-product-line or procurement strategies for their companies, managers need to consider, not only the existence of competing and complementary open source codebases, but also their architecture.