The purpose of this special issue is to provide a rigorous examination of the open source software (OSS) movement, with special consideration to the impact it has on not only the software products, but also the software industry as a whole. Naturally, the OSS movement is large and has many facets that could be worthwhile to investigate. For this special issue, we tried to take as open an approach as possible to soliciting articles pertaining to OSS from diverse perspectives. We were particularly interested in communities and participants in the OSS development process and in the evolving positioning of OSS in the organizational environment. This special issue, as you will see, includes articles answering interesting questions from these two areas.

One common speculation on OSS pertains to the effect of the idealized early model of OSS community influence domains outside of software development. In a recent trade press publication, Tapscott and Williams (2006) lump open source with social network communities (based presumably on their widespread participation and non-cash compensated contributions) and proclaim a social revolution in the making. Ironically, such proclamation may be coming at the same time that OSS communities are being pressured and shaped by market forces to evolve away from the freewheeling social phenomenon toward a more disciplined and organized approach to selling services based on software distributed under relatively open licensing.

Indeed, it has become clear that, over the past decade, open source software has evolved tremendously. As Fitzgerald (2006) points out, the portrait of “supremely talented software programmers who volunteer their services” has become a myth—though perhaps at one time it described most OSS communities. If this mythical portrait is no longer accurate at describing the OSS movement, what has replaced it? Fitzgerald (2006) has termed this more complex portrait OSS 2.0. He characterizes this emergent phenomenon as balancing a “commercial profit value-for-money proposition while still adhering to acceptable open source community values.” In a similar fashion, Watson, Boudreau, York, Greiner, and Wynn (forthcoming) have described the second generation of open source (referred as OSSg2) as a hybrid business model combining characteristics of corporate distribution (where a company provides complementary services around OSS) and sponsored OSS (where a company provides
the majority of the development resources required to create and maintain its products). Such characterizations capture the ongoing transition from the early ‘self-help’ style communities to those very much resembling the corporate structures of a Microsoft or Oracle, only working with services rather than selling code per se. We surmise that, at this point in time, there are OSS groups at many points on the spectrum between these extremes. To the extent that such diverse communities and their members (and their members’ motivations) vary, how does that affect the products created and the ability to position these products in the marketplace? For example, if the OSS model is somewhat dependent on the ‘supremely talented’ programmers selflessly working through the night, would such individuals be equally motivated by working in a more corporate setting responsible for delivering practical solutions to paying clients, rather than selecting the more interesting technical puzzles to work on? By the same token, would corporate programmers take to the rigor, scrutiny, and collective procedures that characterize the ‘mythical’ open source community?

Just as the OSS communities have evolved, the important research questions have broadened and changed. In the early days, a valid research question might have been whether an OSS product had any chance in gaining market share in a given product category. Of course, this question has been answered by software such as Apache and MySQL, which are broadly used. Related technical questions, such as those comparing the performance, reliability, functionality, maintainability, and security of open source code, have similarly been answered in the marketplace. However, where there once may have been general consistency between OSS communities, the numerous transformations of firms and projects in this domain have highlighted the many possible variations for both structure and practices among them. Additionally, relationships between variables at different levels of analysis (such as the quality of code and process vs. the effect on organizations and societies), and the use of a range of existing theoretical bases, both remain to be more fully examined (see Niederman, Wynn, Greiner, Davis, & York, 2006a, 2006b).

Understanding the range and central tendencies of communities, processes, products, and roles in the marketplace becomes an important objective as the number of possibilities in each area increases. And as the range of values for these essential components of OSS domain expand, the interaction between them also grows and a new set of questions emerges. Are there direct relationships between particular community structures or practices (assuming these can be identified) and product outcomes? Are there product categories in which OSS cannot compete effectively with proprietary software? As the range of OSS products has been moving up the stack from technical components rarely seen by an end user (such as Web servers and routers) to components that are manipulated by end users and that incorporate business logic (such as enterprise systems and customer resource management systems), will there come a time when the integration of proprietary business knowledge becomes too entwined with software code to risk its escape through OSS?

This special issue provides additional insight into the OSS phenomenon: whereas the first two articles further improve our understanding of the development process related to OSS (e.g., Crowston and Scozzi, Koch and Neumann), the last three address organizational-level questions associated with issues of adoption (Ven and Verelst), market disruption (Brydon and Vining), and new business models (Feller, Finnegan, and Hayes).

The Crowston and Scozzi (2008) article analyzes four open source software projects to look into the coordination of bug-fixing processes as a proxy for the entire development process. The authors make excellent use of an array of available online data to examine the sequences encountered as various bugs are reported and ultimately closed in the software. In doing so, the authors are able to validate several previously held notions regarding the organizational structure and level of contributions of open source software. The authors also note that many of the traditional coordination
mechanisms found in conventional software development methodologies are not present in open source software projects. This finding highlights and emphasizes one aspect of the distinctive nature of open source software development as compared to conventional development.

The Koch and Neumann (2008) article presents results of the relationship of project processes on outcomes. For some in the MIS research community, process is associated with variables that are observable and that transpire, such as discussions during group meetings or sequences of steps (planned or actual) in a business activity. In this article, the relationship between what we might think of as attributes of the project such as its number of commits or the number of programmers (gross number or as a percentage of the total) who make such commits are referenced as ‘process variables’. These metrics can be viewed as representing a way of inferring different sorts of project processes, even if they do not directly describe or differentiate among the actions involved per se. It is also of interest to consider that Koch and Neumann study open source projects without contrasting these to proprietary projects. To some extent this may be due to the accessibility of data related to open source projects, but it may also represent a “graduation” from justification of open source as a domain of study to observing that open source now represents a significant target for research on its own terms and within its own domain. As noted by the authors, data was collected in a particular country and from a particular type of organization, leaving much room for testing differing reactions to open source for those with differing cultural or organizational values.

Brydon and Vining (2008) highlight how open source has been a major force among infrastructure software products, such as Apache and Linux, largely because of the advanced technical expertise of the user base. They aim to understand whether or not OSS can successfully challenge and disrupt commercial adoption of enterprise software applications (such as CRM and ERP applications) as well as it has for low-level software applications. Building on the concepts of organizational adoptability and community adoptability, the authors develop a middle-range theory and contend that this disruption occurs in two stages, which can be used to explain the trajectory by which the diffusion of these applications can occur. Brydon and Vining employ an interesting methodology by which they compare the results of four ex post case studies of open source software projects to predict the results of an additional ex ante case study with respect to its potential disruptive capacity. The resulting theory presents a viable explanation for why some open source software applications have succeeded and others have not, which may also be of immense benefit for practitioners seeking to have predictive capacity for determining whether or not to adopt a given open source product.

Finally, the article authored by Feller, Finnegan, and Hayes (2008) looks at the
open source phenomenon from the vendors’ perspective, as they consider an alternative business model to create business value. More specifically, these authors focus on a cooperative network of small business partners that find advantages in teaming up to offer a ‘whole product’. They conduct a case study on such an open source cooperative network (e.g., Zea Partners), which, through its 19 small partners dispersed around the globe, seeks to deliver products and services related to a content management system (CMS). This case reveals the agility challenges faced by such a network and contributes to furthering our understanding of the commercialization of OSS in the context of small firms engaged in cooperative business—a business model that we envision becoming more prevalent in the years ahead.

We hope you will enjoy reading the collection of articles in this special issue as much as we enjoyed putting it together. Given open source’s fast transformation, it is exciting to observe new opportunities in both research and teaching. The challenge is ours to take.

REFERENCES


