2-2002

Institutional Entrepeneurship in the Sponsorship of Common Technological Standards: The Case of Sun Microsystems and Java

Raghu Garud
New York University

Sanjay Jain
University of Wisconsin - Madison

Arun Kumaraswamy
West Chester University of Pennsylvania, akumaraswamy@wcupa.edu

Recommended Citation

Garud, R., Jain, S., & Kumaraswamy, A. (2002). Institutional Entrepeneurship in the Sponsorship of Common Technological Standards: The Case of Sun Microsystems and Java. *Academy of Management Journal, 45*(1), 196-214. http://dx.doi.org/10.2307/3069292

© Academy of Management Journal
2002, Vol. 45, No. 1, 196–214.

INSTITUTIONAL ENTREPRENEURSHIP IN THE SPONSORSHIP OF COMMON TECHNOLOGICAL STANDARDS: THE CASE OF SUN MICROSYSTEMS AND JAVA

RAGHU GARUD
New York University

SANJAY JAIN
University of Wisconsin—Madison

ARUN KUMARASWAMY
Rutgers University, Camden

The institutional entrepreneurship implicit in a firm's sponsorship of its technology as a common standard is beset by several challenges. These challenges arise from a standard's property to enable and constrain even as potential competitors agree to cooperate on its creation. Our exploration of Sun Microsystems's sponsorship of its Java technology suggests that standards in the making generate seeds of self-destruction. Our study also identifies the social and political skills that a sponsor deploys to address these challenges.

Institutional theorists draw attention to the taken-for-granted facets of social and economic life (cf. Powell & DiMaggio, 1991; Scott, 1995). These taken-for-granted facets represent institutionalized rules that manifest themselves in the ways that people frame issues, make choices, and pursue behaviors (Jepperson, 1991). Indeed, conformity to institutionalized rules may generate path dependence leading to specific ways of thinking and doing (Arthur, 1989; David, 1985; North, 1990).

Although considerable work exists on organizational compliance to preexisting institutions, relatively little work explores how these institutions arise in the first place (Fligstein, 1997). As DiMaggio observed: "Institutional theory tells us relatively little about 'institutionalization' as an unfinished process (as opposed to an achieved state)" (1988: 12). An appreciation of how institutions arise can add to understanding of how and why people operate in today's environment relying on institutions hammered out in the past. It can also add to understanding of how institutions can be created for the future as new technological, geographical, and cultural imperatives emerge.

Of specific interest to us is the role of actors in shaping emerging institutions (Christensen, Karnøe, Pedersen, & Dobbin 1997; Scott & Christensen, 1995). In this regard, Fligstein (1999) offered two ways of bringing actors back into a theory of institution creation. One is to borrow from economic theories, such as game theory and other models of rational action. Such a view discounts the messy, political processes involved and can easily lead to a post hoc, rationalized view of how institutions emerge. A second way is to employ the tenets of institutional theory in organizational analysis and explore how actors build their goals and procedures directly into emerging institutions (Hirsch, 1975; Meyer & Rowan, 1977). We pursued this perspective in this study focusing on the "skilled performances" of social actors that lie at the core of the production and reproduction of social life (Giddens, 1979).

Initiatives to shape institutions as they emerge represent acts of institutional entrepreneurship (DiMaggio, 1988; Fligstein, 1997; Selznick, 1957). Institutional entrepreneurs create a whole new system of meaning that ties the functioning of disparate sets of institutions together (DiMaggio, 1988). They define, legitimize, combat, or co-opt rivals to succeed in their institutional projects (Scott, 1994). Assuming the role of champions, they energize efforts toward collective action and devise strategies for establishing stable sequences of interaction

The alphabetical ordering of authorship reflects the fully collaborative nature of this work. We thank the people of Sun Microsystems for their help with the Java case. We also gratefully acknowledge Roger Dunbar, Paul Hirsch, Huggy Rao, Dick Scott, Persephone Doliner, the anonymous reviewers of *AMJ*, and participants in the 19th Strategic Management Society conference, Berlin, 1999, for their helpful feedback.

with other organizations to create entirely new industries and associated institutions (Aldrich & Fiol, 1994).

These acts of institutional entrepreneurship are becoming increasingly important as new technologies break open taken-for-granted assumptions that constitute the institutional "black box." No technology exists in a vacuum. Each requires a defined institutional space with rules that govern the production, distribution, and consumption of associated artifacts (Dosi, 1982; Rosenberg, 1982; Van de Ven & Garud, 1994). Indeed, technological fields are embedded in the institutional environments that shape them (Dacin, Ventresca, & Beale, 1999; Garud & Jain, 1996).

Technological standards are key facets of this institutional space (Jain, 2001). They represent interface specifications or "rules of engagement" that dictate how different components of technological systems work together to provide utility to users (Garud & Kumaraswamy, 1993). Often, common technological standards have to emerge before users can evaluate and exchange products in the marketplace (Garud & Rappa, 1994). In other words, common standards offer a framework within which product-markets operate (Garud & Karnøe, 2002; Porac, Rosa, Spanjol, & Saxon, 2001).

By shaping common standards, firms can build attributes of their technologies directly into emerging institutional structures (Constant, 1980). This is consistent with a Lamarckian conceptualization of human agency in which agents directly shape the selection mechanisms that then govern their functioning (Gould, 1980). Firms can derive significant competitive benefits by successfully shaping common standards (Hamel & Prahalad, 1994). This proposition is particularly true of information technology fields characterized by network externalities and increasing returns (Arthur, 1989; Farrell & Saloner, 1986; Katz & Shapiro, 1985; Shapiro & Varian, 1999). In such network technological fields, mutualistically interdependent firms produce individual components of larger technological systems (Barnett, 1990; Garud & Kumaraswamy, 1995a; Langlois & Robertson, 1992).

We explore the challenges that an individual firm faces in sponsoring its own technology as a common standard within network technological fields. To address this facet of institutional entrepreneurship, we carried out a deep exploration of Sun Microsystems' efforts to sponsor its Java software technology. Introduced by Sun in 1995, Java enables computers and information appliances to run applications distributed over a network. Barely seven years old, Java has emerged as a common standard, although its ownership has been contested and its eventual success was not assured.

Sun's role in sponsoring Java is illuminating. The firm has attempted to set and shape a standard before product-markets have emerged fully. Sun has done this by offering others access to its proprietary technology, thereby starting a bandwagon effect around Java. However, Sun has also controlled access to the technology on occasions when it thought that Java had become "too hot." Indeed, Sun's sponsorship of Java is a story full of contradictions reflecting the wider dilemmas surrounding sponsorship of technology through institutional processes.

To explore these contradictions and dilemmas, we begin the article with an overview of technological fields and the role that standards play in their governance. Then, we discuss tensions that are inherent in the creation and maintenance of common standards and the challenges that a firm confronts in sponsoring common standards over time. Subsequent to our exploration of how Sun has been confronting these challenges in the case of Java, we offer insights on institutional entrepreneurship within emerging network technological fields.

TECHNOLOGICAL FIELDS AND STANDARDS

Technological fields represent a pattern of relationships among objects and humans related to a product-market domain (Callon, 1986; Garud & Karnøe, 2002). This conceptualization is similar to the notion of organizational fields comprising a shared set of meanings (Scott, 1995: 130). Within technological fields, meanings of artifacts and patterns of interaction among actors emerge through a negotiated process (Bijker, Hughes, & Pinch, 1987).

Of specific interest to us is the institutional environment that governs the patterns of interaction within technological fields. This institutional environment includes public policy regimes (Dobbin & Dowd, 1997), regulatory instruments (Van de Ven & Garud, 1989), and mechanisms for venture capital financing (Kenney, 2000; Suchman, 1994). It also includes different sources of legitimacy (Aldrich & Fiol, 1994; Rao, 1994) and underlying norms of community interaction (Karnøe, 1999).

A rich stream of research on the social construction of technology explores processes underlying the emergence of institutions. For instance, new technological fields have to establish legitimacy to generate momentum (Hughes, 1983). These legitimacy battles are manifest not only in clashes between old and new technological fields, but also in clashes between alternative technological trajectories within a field, as each vies to become the "dominant design" (Garud & Karnøe, 2002; Tushman & Anderson, 1986; Utterback & Abernathy, 1975). Indeed, new evaluation criteria have to emerge and be recognized for a new field to establish its legitimacy (Constant, 1980). Competition among technologies occurs both between and within evaluation criteria (Garud & Rappa, 1994). Eventually, institutional closure brings about stability around a set of specific evaluation criteria (Van den Belt & Rip, 1987). Once this happens, these institutional facets represent "traces of order in man-made complexities" (Hughes, 1983: ix) that imbue a technological field with a "distinctive style" that reflects the creative latitude of the actors who contributed to its creation (Hughes, 1983).

At the heart of these technological fields lie technological systems. Technological systems comprise a set of components that interact with one another to provide utility to users. A system's performance is dependent not only on the performance of constituent components but also on the extent to which they are compatible with one another (Garud & Kumaraswamy, 1993; Henderson & Clark, 1990; Jain, 2001; Sanchez, 1995; Schilling, 2000). Compatibility among system components is achieved by designing them to common standards. Standards are codified specifications that detail the form and function of individual components and the rules of engagement among them. Together, specifications about the components' form and function and the rules determining their interaction define a system's "architecture" (Ferguson & Morris, 1993).

The presence or absence of common standards impacts innovation within a technological field. Rigid adherence to common standards may dampen innovation. This is because interdependent firms find it difficult to innovate outside the common standard for fear of introducing incompatibilities. On the other hand, absent common standards, firms that develop disparate components of a technological system are unable to coordinate their activities easily (cf. Brunsson & Jacobsson, 2000). Consequently, components manufactured by different firms may not interact or perform well with one another.

These observations arise from a paradoxical property of common standards—they enable and constrain at the same time (Garud & Jain, 1996; Jain, 2001). This "structurational" property of standards is representative of a broader proposition that structure is both medium and outcome of action (Giddens, 1984). In other words, structures generated to enable action also begin to constrain it. A standard too can be viewed as both medium and outcome of firms' actions. They enable by providing producers and users an opportunity to develop or use different parts of a technological system in a distributed manner. At the same time, they constrain its evolution to certain directions.[1]

This structurational property of standards creates a problem in mobilizing a coalition around a new standard. We know that free-riding behavior dampens collective action (cf. Olson, 1965). The enabling and constraining property of standards suggests additional challenges in mobilizing a coalition. First, new technologies have to overcome inertia to supplant standards enabling the functioning of existing technological fields. Second, firms may realize that agreement on a standard may constrain their activities in the future.

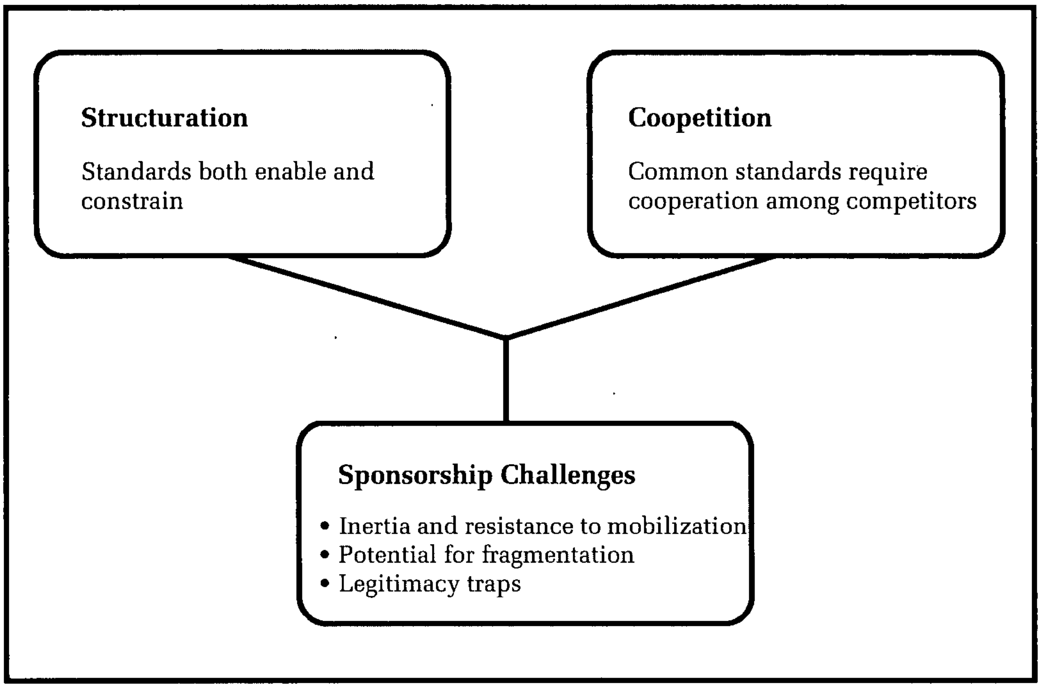
The problem of generating collective action around a new standard becomes even more pronounced when the emerging coalition comprises firms, which have private interests. This problem arises from the presence of simultaneous cooperation and competition, which usually lie in contrast with one another. Yet, in generating collective action to create common standards, cooperation has to be induced among a group of firms, some of which may be rivals.

This "coopetitional" (Brandenburger & Nalebuff, 1995) property of standards ensures that cooperation among members of a standards collective is uneasy at best. Inducing and maintaining such a collective is a challenge even for a neutral body. It is even more of a challenge for a firm that, in its own private interest, wants to sponsor its technology as a common standard. This is because a sponsoring firm has to convince potential competitors to support a common standard that could offer the sponsor a competitive advantage in the future. Complicating the situation is the active resistance that a sponsor is likely to confront from incumbents in established technological fields whose dominance is threatened by the sponsor's initiatives.

Our purpose in offering these observations is to draw attention to the tensions that structurational and coopetitional properties create during the standardization process (see Figure 1). These tensions, in turn, create challenges for a firm attempting to sponsor its technology as a common standard. We examined Sun's efforts to sponsor its Java technology to gain a finer appreciation of what it means to

[1] Just as organizational fields may be "overembedded" or "underembedded" in a system of social relations (Granovetter, 1985; Uzzi, 1996), so too can technological fields be associated with different levels of embeddedness based on the extent to which a standard enables or constrains (Garud & Jain, 1996).

FIGURE 1
Standards in the Making: Sponsorship Challenges



be an institutional entrepreneur in an emerging technological field.

RESEARCH SITE AND METHODS

We chose to study Sun's sponsorship of its Java technology as a common standard for several reasons. First, Java is a software technology that exemplifies the distributed computing paradigm characteristic of network technological fields. Second, Sun had had prior experience and success in technology sponsorship, most notably in the workstation market (Garud & Kumaraswamy, 1993). Third, data on Sun's role in sponsoring Java are readily available. All these factors make the Java technological field a strategic research site (Bijker et al., 1987).

What makes this research site particularly interesting is that Sun struggled in its sponsorship efforts despite its considerable experience in sponsoring common standards and its possession of what most analysts consider a revolutionary technology in Java. Indeed, at various stages of Java's evolution, Sun's success was far from assured. Consequently, our study of this emerging technological field was consistent with the position that it is important to view success and failure symmetrically (Bijker et al., 1987). Studying an emerging technological field also compelled us to conceptualize institutional entrepreneurship in process rather than in variance terms (Mohr, 1982), leading to insights that recognize the precariousness of such endeavors.

Our objective was to offer observations on the challenges of technology sponsorship based on systematic data analysis. Technology sponsorship efforts are complex initiatives embedded within coevolutionary dynamics (Rosenkopf & Tushman, 1994; Van de Ven & Garud, 1994). This complexity renders the study of technology sponsorship amenable to naturalistic inquiry in which insights are induced through interpretive means (Lincoln & Guba, 1985). Specifically, the unfolding of these processes is better explicated by tracing their historical roots using inductive logic.

The aim of naturalistic inquiry is to generalize from a case to a theory rather than from a sample to a population. Typically, this is accomplished by "iterating" between data and theoretical constructs until a stage of theoretical saturation is reached (Glaser & Strauss, 1967). Moreover, this inquiry mode emphasizes "procedural adequacy" and "credibility" (Lincoln & Guba, 1985), which we established by employing steps that Miles and Huberman (1984) suggested in their primer on qualitative research.

We tracked publicly available information from several on-line technology news services, including CNET News, New York Times Online, San Jose Mercury News Online, TechWeb, the Wall Street

TABLE 1
Java Standardization: Chronology

Year	Major Events
1995	Sun announces Java commercially on May 23, 1995. Sun allows developers to download Java from its Web site for free. By year-end, Sun has licensed Java to 38 vendors, including IBM, Oracle, Intel, and AT&T. Microsoft licenses Java from Sun in December.
1996	Sun announces JavaSoft, a subsidiary to administer Java-related activities. Sun announces initiatives to develop new technologies to increase Java's functionality. Kleiner Perkins Caufield Byers creates a $100 million venture fund to finance Java start-ups. Sun releases a more robust version of Java, JDK 1.1. Sun introduces stringent Java compatibility tests and the 100% Pure Java initiative. By year-end, Sun's list of Java licensees has grown to over 100 vendors.
1997	Sun announces delivery of new Java technologies by the end of 1997. Sun applies to the International Organization for Standards (ISO) to get Java accepted as a global standard. Sun sues Microsoft for infringing its Java licensing agreement. By year-end, the number of Java licensees has increased to over 150.
1998	Sun licenses Java to TCI and Motorola for use in mass-market consumer products. Hewlett Packard announces plans to market its own embedded Java clone. Sun announces broad joint Java development pact with IBM. Sun announces reorganization of its Java subsidiary to separate its Java applications development effort from its Java standardization effort. Sun also announces that PriceWaterhouseCooper will monitor the standardization process. Sun announces delays in delivering Java technologies such as JDK 1.2 and HotSpot. Sun acquires NetDynamics, a vendor of Java-based enterprise servers. Hewlett Packard's efforts to get its alternative specifications accepted as the embedded Java standard fails. Sun sponsors an industry-wide effort led by IBM to develop embedded Java specifications. Sun gains a victory in its license-infringement lawsuit against Microsoft. Sun announces a joint Java development and marketing alliance with AOL-Netscape. Sun announces new community-source licensing model for Java 2 platform. By year-end, the number of Java licensees has increased to over 200.
1999	Sun releases the Java 2 platform. Sun introduces Jini, a Java-based networking technology for consumer appliances. Sun releases portions of the Java source code. Sun announces that major consumer electronics firms support use of Java as a digital television platform. Sun announces a partnership with 14 vendors to create an open interface technology to connect home and small business appliances. Microsoft and Hewlett Packard join several other companies to form the J-Consortium, a group that aims to establish an alternate standard for Java. Sun switches from the ISO to the European Computer Manufacturers Association (ECMA) in its attempts to make Java an official standard. Transvirtual Technologies, a company funded by Microsoft, announces Java clone. Alan Baratz, head of Java software development efforts, resigns; Sun appoints Pat Sueltz, formerly the head of IBM's Java efforts, to take his place. Sun abandons attempt to standardize Java through the ECMA in December. Sun starts charging Java vendors a branding fee for products based on Java 2 platform.
2000	Sun offers source code for the Java 2 platform, Standard Edition, free of charge. Sun announces the Joint Community Process 2.0, a more open process for the future development of Java.

Journal Interactive Edition, and Wired News, on a real-time basis. We also downloaded all articles on Java published between 1993 and 1998 in three important trade journals—*Computerworld, PC Week,* and *Infoworld.* We gained access to press releases, white papers, and articles pertaining to Java from Sun's Java Web site *(java.sun.com).* These company documents offered us information on various aspects of the Java technology, milestones in Sun's sponsorship efforts, and Sun's licensing policies with respect to Java.

We used these data to generate a chronology of events; Table 1 is an abbreviated list of these events.[2] The multiple data sources helped us accomplish "triangulation" (Jick, 1979). There were very few disagreements among the data sources on the factual details pertaining to Java's emergence and evolution. However, we noticed that Sun's actions often appeared to be contradictory over time,

[2] This case covers events until April 2000.

reflecting the challenges involved in technology sponsorship.

As we developed the chronology and the case, we were cognizant of the theoretical issues and constructs that emerged from the data. In ongoing discussions, we explored tentative constructs and relationships among them to address pertinent questions. We actively abandoned or modified tentative hypotheses and retained those that had greater validity in view of the stream of data confronting us daily. We kept iterating between data and theory till theoretical saturation was reached. Indeed, recent additional events pertaining to Sun's sponsorship efforts have only served to confirm the validity of our framing and the focus of our study.

We grappled with the level of generality at which we wanted to articulate our findings, deciding to articulate a theory that would be specifically applicable to network technological fields, although our observations may be generalizable to other technological fields as well. Studying phenomena in the making requires teasing out the generative mechanisms that underlie institutionalization processes (Tsoukas, 1989). This perspective, when applied to Sun's sponsorship of Java, offered two insights—one at the level of standards and the other at the level of the sponsor. We found that standards in the making generate seeds of self-destruction (cf. Wade, 1996). At the sponsor level, Sun's actions suggest that such attempts are fraught with failure and that a sponsor has to deploy a combination of social and political skills in a mix that depends upon the exigencies of the situation. We develop these insights in the rest of the article.

JAVA: A STANDARD IN THE MAKING

A Solution Looking for a Problem

In the early nineties, a small group of engineers at Sun Microsystems created an ambitious software program called Oak to enable various electronic devices to run applications distributed to them over a network. Initial attempts to commercialize Oak—first for interactive TV, then for video game players, and finally for multimedia CD-ROM development—did not succeed. Subsequently, Oak's original creators adapted the software program to run on networks such as the Internet. In January 1995, Sun renamed Oak as Java and decided to offer it as a new programming environment for the Internet.

Sun saw in Java an opportunity to position itself as a leader driving the Internet revolution. In creating a new technological field around Java, Sun would be able to break away from the increasingly marginalized Unix field as well as to counter the increasing dominance of the Windows technological field. If Java succeeded and Internet use grew, Sun would be able to shift the emphasis of the entire information technology field toward a network-centric approach (its key strength) and thereby validate its long-held belief that "the network is the computer." In other words, Sun would be able to leverage its ownership of Java to transform itself into a central player within the wider post-PC, Internet-based information technology field. Therefore, making Java an architectural standard for the Internet was important for Sun.

However, Sun's uncontested window of opportunity was small. Microsoft had announced its intention to release a comparable technology called Blackbird for its Windows platform in early 1996. Given Microsoft's dominance, there was a possibility that Blackbird would become a dominant standard, thereby further strengthening the entrenched Windows technological field. Additionally, consistent with Schumpeter's (1942) notion of potential competition from future technologies, there was a threat that other firms would eventually create comparable technologies. Therefore, despite the absence of any other comparable technology in the market at that time, Sun had to move quickly to gain support for its fledgling technology.

Sun's Strategic Actions as a Sponsor

In network technological fields, a powerful inducement for a firm to subscribe to an emerging standard is access to valuable technology that might otherwise take considerable time and resources to develop. This observation translates into an "open systems" strategy for the technology sponsor, a strategy that Sun had successfully pioneered earlier in the computer workstation market (Garud & Kumaraswamy, 1993). An open systems strategy allows both rivals and vendors of complementary products easy access to the sponsor's proprietary technology.

For a sponsor, adopting an open systems strategy implies placing part of its private knowledge in the public domain. A property of such public goods is that even those who have not contributed to their creation can benefit from them (Olson, 1965). Typically, such a situation creates a free rider problem resulting in the underdevelopment of the public good or in the degradation of the "commons" (Hardin, 1982). However, in network technological fields that have the potential to exhibit increasing returns (Arthur, 1989), the conversion of some portion of private goods into public goods can attract others to join a collective (Raymond, 1999). Others joining increases the carrying capacity of the emerging technological field as the firms joining the collective develop complementary assets, thereby generating momentum (Hughes, 1983). The excess value that is created compensates firms for their willingness to cooperate.

Sun took advantage of such open systems dynamics to generate momentum behind its Java technology. First, Sun allowed third-party developers to download Java for free from its Web site. Second, it emphasized the "write-once, run-anywhere" capability of Java, contrasting this feature with the typical need to rewrite software for each major platform. Sun also made it easy for software developers to learn and use Java with initiatives such as the creation of Java development tools, courses, and partnerships (Sun Microsystems, 1999).

Often, an open systems strategy goes together with a strategy of "priming" future expectations (Farrell & Saloner, 1986; Garud, Jain, & Phelps, 1998). For firms subscribing to an emerging standard, benefits do not exist in the present and may be realized only if others too are persuaded to subscribe to the standard. In such scenarios, the technology sponsor attempts to generate such expectations by preannouncing access to future technologies even when product-markets have not yet emerged. If a sponsor is successful in its agenda-setting efforts (Lukes, 1974), interdependent firms may be persuaded to join the bandwagon behind the emerging standard. As the bandwagon grows, so does the standard's legitimacy (Wade, 1995). Eventually, this enhanced legitimacy results in the deployment of technical and financial resources that, in a self-fulfilling manner, generate the promised value from cooperation.

Sun began priming expectations by aggressively marketing the Java brand and offering a vision of what Java would eventually become—a complete networking platform. Initially, however, this vision was far beyond Java's limited functionality. As a result of these agenda-setting efforts, potential partners and vendors began subscribing to Java and awaited release of Java enhancements. As George Paolini, director of corporate marketing at Sun's Java Soft subsidiary, acknowledged, Sun's fight was for the minds of individual developers and potential partners:

> In today's world, it's really about first creating mindshare and awareness about a technology, and then driving that technology to reality. That's really what Java has been about. (Kirsner, 1997)

One can also view these initiatives as efforts by Sun to establish momentum behind the emerging Java technological field. These efforts became more focused as Sun began promoting Java as an initiative mobilized against Microsoft's desktop-centric view of computing. Indeed, Sun's CEO, Scott McNealy, framed this mobilization of the "Java force" against Microsoft as a metaphor evoking *Star Wars:*

> There are two camps, those in Redmond, who live on the Death Star, and the rest of us, the rebel forces. (Surowiecki, 1997)

Sun's slogan, "The network is the computer," and its promise to make Java a write-once, run-everywhere platform had an intuitive appeal for programmers hoping to write software for the Internet. Clearly, the battle was for the minds of users and vendors as much as it was for their computing preferences.

Consistent with Fligstein's (1997) observations on how an institutional entrepreneur mobilizes support, this strategic framing helped Sun gain the support of a broad set of partners, including systems assemblers, software firms, and component manufacturers. By the end of 1995, nearly 40 key vendors, including Adobe, AT&T, Borland, IBM, Intel, Oracle, Symantec, and Toshiba, had licensed Java from Sun. Also, major venture capital firms, such as Kleiner Perkins Caufield Byers, had jumped on to the Java bandwagon by setting up venture funds to finance new companies creating Java-based applications (Schlender, 1997). As an increasing number of developers and vendors adopted Java, momentum grew rapidly. Not surprisingly, the vendors that had the most to gain from displacing Microsoft's dominance were among the first to join the Java field.

Eric Schmidt, then Sun's chief technology officer, suggested that Sun was trying to create an architectural franchise around its Java technology (Bank, 1995). However, Sun and members of the emerging Java field realized that they would have to compete with one another eventually and that Sun would attempt to consolidate its own position in the product-market even as it sponsored Java as a common standard. As George Paolini observed:

> Our model has always been to agree on a platform and compete on its implementation. (Nerney, 1998)

A typical Java licensing agreement for commercial use sought an up-front fee and royalties on unit sales of Java-based products. But it allowed licensees to modify the technology as long as they shared these modifications freely with Sun and other licensees. Such a liberal policy would provide licensees with the flexibility to innovate, thereby enabling the technology to evolve faster than Sun's solitary efforts could ensure. Also, it would allow members of the Java field the flexibility to interpret the technology to suit their own purposes (Pinch & Bijker, 1987) and differentiate their product offerings from one another and from Sun's own offerings. Such flexibility during Java's early stages was key to the mobilization of support from vendors who otherwise would have had to develop comparable technologies on their own.

Countermobilization by Microsoft

An open systems strategy requires that the technology be equally accessible to all, both competitors and collaborators. An irony of the open process is that powerful incumbent firms may withhold support by not joining the bandwagon developing around the emerging standard. After all, why should they "credit" (Weick, 1979) an initiative that could fundamentally challenge and compromise their dominance? This was the case with Java, too.

Microsoft's early reaction was to actively ignore Java and concentrate on promoting and developing Blackbird, its alternative software. For Sun, Microsoft's refusal to endorse Java posed a legitimacy challenge from the dominant Windows technological field. Later, as Java began gaining momentum, Microsoft began discrediting Java. Microsoft claimed that Java was not really revolutionary and that enhancements planned for its own proprietary Visual Basic programming language would offer programmers as much functionality and flexibility as Java (Wingfield, 1995).

This cat-and-mouse game between Sun and Microsoft came to a head when Microsoft finally capitulated and decided to license Java after Sun paraded the results of its successful mobilization efforts at a trade show in December 1995 (Elmer-Dewitt, 1996). At this stage, Microsoft's Web browser was falling behind Netscape's because it lacked the capacity to function with Java. In other words, Microsoft could no longer afford not to endorse Java without being left behind.

There was some ambiguity as to whether Microsoft's announcement to license Java was an endorsement or a threat. Both were probably implied, reflecting the coopetitional nature of standardization initiatives. As one industry analyst wrote:

> Everybody expected Microsoft to strike back, reaffirming its commitment to its own Java-like Visual Basic. But at the last minute, Gates changed his mind, announcing that he too would license Java, while also promising somewhat menacingly to "extend" it. (Elmer-Dewitt, 1996)

And extend Java, Microsoft did. Taking advantage of Sun's inability to deliver on its vision and commitment to improve Java's functionality (Nerney, 1997), Microsoft began adding proprietary extensions to Java to improve how it worked with the Windows operating system (Rein, 1997). Sun became concerned that Microsoft was trying to "poison" Java by creating its own, incompatible version. If Java fragmented into incompatible versions, its write-once, run-anywhere capability would be compromised, thereby reducing the incentive for vendors and developers to subscribe to Java. Moreover, Sun would lose control of the software and face credibility problems in creating a unified technological field around it. Sun had earlier seen fragmentation within the Unix technological field and wanted to prevent a similar occurrence with its Java technology.[3]

Also at stake here was a battle for the very meaning of Java within the wider information technology field. Technological systems can be viewed as being organized in a hierarchy, with some components being more central than others (Clark, 1985; Hughes, 1983). Events in the Java technological field suggest that this hierarchy itself is not a given but is hotly contested. In licensing Java, Microsoft attempted to subsume it within its existing portfolio of technologies. In this regard, whereas Sun referred to Java as the applications platform for the Internet, Microsoft sought to portray it merely as one of many programming languages that it employed, thereby downplaying its significance.

In sum, Sun wanted Java to be at the apex of the hierarchy, whereas Microsoft wanted its Windows operating system to remain at the apex. The stakes were high as the emerging hierarchy would critically impact the pecking order of firms not only within the Java field but also within the wider information technology field. Appreciating the high stakes involved, Sun decided to challenge Microsoft's actions in the courts and wage a protracted legal battle. In 1997, Sun filed a lawsuit charging Microsoft with infringement of the Java licensing agreement.

[3] In the case of Unix, AT&T, restricted from entering the computer industry by regulators, had licensed its Unix operating system technology liberally to all vendors. Over time, licensees had introduced their own proprietary technologies into the licensed version, thereby creating several incompatible versions of Unix. This fragmentation had prevented Unix from becoming dominant in the computer industry. Competitive pressures and diverging interests have continued to plague attempts by competing factions to unify Unix.

Sun Attempts to Rein in Java

Besides waging a legal battle with Microsoft, Sun simultaneously initiated other actions to prevent fragmentation of its emerging Java standard. For instance, Sun introduced an extensive suite of over 5,000 compatibility tests in December 1996. Soon thereafter, Sun announced the 100% Pure Java initiative, under which Sun or its agent (KeyLabs Inc.) would certify Java applications that passed its stringent compatibility tests. In a statement released on December 3, Jon Kannegaard, vice president of Sunsoft, Sun's software division, said:

> There is nothing more important to us than a ubiquitous, compatible Java platform. . . . We've devoted enormous resources to developing robust and demanding tests to ensure that Java does indeed do what we say it will do—in every implementation, on every platform.

Although Sun's actions in enforcing Java compatibility primarily sought to curb Microsoft, they also alerted other partners to Sun's seriousness about protecting its Java franchise. As Alan Baratz, then JavaSoft's president, proclaimed:

> There is a package of Java functionality that we deliver to the licensees. None of our licensees are allowed to add, delete or modify things in that package. (Helft & Mardesich, 1997)

Several vendors, primarily Microsoft and its supporters (such as DEC, Intel, and Compaq), advised Sun to hand over the administration of Java to an international standards body (Dow Jones Newswires, 1997). Mr. McNealy, Sun's CEO, declined, and he reiterated Sun's intention to use an "open-control" model of standardization wherein it would keep the technology open but retain enough control to drive the standard forward without waiting for consensus to develop (Jones, 1997). In explaining Sun's actions, Jim Mitchell, then vice president of JavaSoft, stated:

> Java is a brand name with its own value and integrity that Sun must maintain. . . . We'll put Java in the public domain when Microsoft gives up Windows. (Gage, 1997)

Nevertheless, to formally legitimize its role as Java's sole steward, Sun decided to seek the involvement of a neutral institutional body in its Java sponsorship effort. Specifically, in March 1997, Sun approached the International Organization for Standardization (ISO), the premier standards organization in the world, to gain recognition for Java as a publicly available specification (PAS). Such an initiative, if successful, would confer on Java the status of an open international standard and simultaneously protect Sun's control over Java's evolution.

Sun's open-control model for administering Java illustrates the tension inherent in technology sponsorship. To enable Java's evolution into a technology that justified its original promise, Sun had to allow members of the collective to adapt it for their own use. At the same time, Sun had to exercise control to ensure that the technology was not compromised by the creation of incompatible versions. Such control was also necessary to mediate effectively between members of the collective when disputes arose over interpretation and implementation of the technology.

Concerns Arise about Sun's Credibility

As Java began gaining in popularity, several members of the Java collective became concerned over what they perceived as Sun's excessive control over Java. Such concerns were heightened by Sun's introduction of Java products that competed with those offered by other members of the Java collective. Even ardent supporters were afraid that Sun's control would give it undue advantage when competition intensified in the Java product-market. Pat Sueltz, then general manager of Java software at IBM, suggested:

> Sun . . . should establish the standard and compete above it. To the extent that Sun has any advantage, it limits the creativity of their partners. (Moeller, 1997)

Many small vendors also felt that Sun was biased in its stewardship of the emerging Java standard. Depositions in the Sun-Microsoft court case revealed that Sun had secretly granted waivers to favored partners (such as IBM, Novell, and Spyglass) that had failed its Java compatibility tests (Helft & Quinlan, 1998) even as it used strong-arm tactics against less-favored ones. In another case alleging bias, Rick Ross, president of a small firm, Activated Intelligence, complained that Sun had broken a promise to include his firm's code in the Java specification, in favor of code written by Kodak, a larger and hence more valuable partner (Oakes, 1998). These small vendors began perceiving Sun's control over Java as a case of a new tyrant (Sun) trying to replace the old one (Microsoft).

These events illustrate the intensely political nature of the sponsorship process. They also offer key insights into the complex roles played by a technology sponsor. Initiatives required to enable the creation of a standard (opening up) are different from those required to enforce compatibility (reining in). To the extent that the technology sponsor takes on the dual responsibility of creating as well as enforcing the rules, there is potential for a loss of credibility. As Bill Roth, product manager of Java at Sun, pointed out in an interview:

> We're doing our best to be faithful stewards, but there are folks who would love to wrest control of Java away from us. (Shankland, 2000a)

Compounding this problem is the potential for members of the collective to view the sponsor as departing from its earlier publicized vision. Sun had mobilized a coalition around Java by framing issues in terms of the benefits that would accrue to all potential partners. With time, as the technology matured and competition intensified in the emerging product-market, Sun was perceived as reframing the issues and changing the rules of the game in its own favor. Indeed, as a member of the Java collective critical of Sun's role complained:

> They want to be the referee, but they want to play in the game too. (Nerney, 1998)

Rifts due to Coevolutionary Dynamics Emerge

Sun's actions as a rule enforcer began compromising its credibility to such an extent that even close partners such as Netscape and Novell advised Sun to make Java "open-source" software, albeit with some safeguards to ensure its beneficial evolution (Gilmore, 1998). Prompted by concerns over Sun's stifling control, Hewlett Packard (HP) announced plans in March 1998 to market its own embedded Java clone in competition with Sun's "favored" version. According to Jim Bell, the leader of HP's Java development team:

> Java is so important that we feel that it has outgrown the ability of one company to control it. (Gomes, 1998a)

To deflect criticism of excessive control, Sun created an industry-wide group to develop an embedded Java standard under the National Institute of Standards and Technology (NIST). However, HP and several other vendors were suspicious that Sun would co-opt this supposedly unbiased group into adopting its own embedded Java specifications as the standard. Indeed, Kelvin Nilsen, chief technology officer at NewMonics, a vendor involved in the embedded Java standardization process, expressed this suspicion:

> I think Sun would like to take the open NIST process and bring it under their wing. . . . There's also a sense Sun may leave the (NIST) group as we move forward. (Wolfe, 1998)

To prevent Sun from deriving an unfair advantage in the embedded Java product-market, HP and its supporters offered a proposal that competed with NIST. Although Sun eventually defeated this proposal by brandishing the unwelcome prospect of fragmentation (Sliwa, 1999a), this event formalized the rift within the Java collective. HP and its supporters decided to pursue further work on their own version of embedded Java. Microsoft, still constrained by Sun's legal action, promptly licensed HP's version of Java for its own use. Reflecting upon these coevolutionary dynamics that resulted in partners spinning off to work on their own initiatives, an industry analyst commented:

> Indeed, Sun's relationship with Java licensees has been fractious. Last year, several licensees, including HP, split off to form their own standard for real-time Java. Even huge partners like IBM and Novell chafed under Sun's control of the standards process. (Darrow, 2000)

Sun Attempts to Regain Credibility

Sun sought to address concerns that it was trying to consolidate its own position in the emerging Java product-market at the expense of other members of the Java collective. It reorganized its Java business to separate standardization efforts from product development (Clark, 1998). While announcing third-party auditing of its stewardship of the Java standard to ensure fairness, Sun also promised to share its Java product introduction plans with members of the Java collective.

In December 1998, Sun changed its licensing model to provide more flexibility to its partners in the Java collective. It began licensing the second generation of Java technology under a community source licensing model, under which Sun made portions of the Java source code publicly available (Gomes, 1998b). As in the other emerging open-source initiatives in the information technology field, any licensee could adopt and modify Java without sharing the modifications with Sun. Licensees could also share their modifications with one another without getting prior permission from Sun. Furthermore, even Java clone makers could use the Java brand name for a fee as long as they included certain Sun technologies in their products (*Computerworld,* 1998). The only constraint that Sun imposed on licensees and clone makers was that their Java-based products had to pass Sun-defined compatibility tests. Despite these concessions, however, Sun was not yet ready to relinquish control. As Mr. Mitchell, vice president at Sun's Java software division, cautioned:

> But, we are not going to go overboard [in opening the Java standards process] because the day we do that, the big guys will come out, and they can spend lots of money, and [there] will be fragmentation (Sliwa, 1998).

Formal Legitimacy Eludes Sun

Notwithstanding its attempts to regain credibility, Sun was dealt a setback in its bid to get Java certified by the ISO as a PAS. After protracted negotiations spanning two years, ISO members voted to require sponsors of a PAS to cede maintenance of the standard to the international body. Sun immediately backed away from its initiative to make Java a PAS under ISO's auspices.

Instead, in May 1999, it approached the European Computer Manufacturers Association (ECMA), another important standards organization, to seek their endorsement for Java (Bingley, 1999). At the ECMA too, Sun faced opposition to its insistence on retaining control over Java. After many fractious meetings with members of the ECMA, Sun decided to discontinue its efforts to seek ECMA endorsement (Sliwa, 1999b). The association's executives and members found Sun's actions so egregious that Jan van den Beld, the secretary general of ECMA, complained in a public letter:

> Their action over the past two years has resulted in an enormous waste of experts' time and companies' money. (Shankland, 2000b)

In sum, Sun's aborted attempts at gaining legitimacy instead led to depletion of legitimacy. A sponsor's legitimacy is destroyed to the extent that it attempts to co-opt a neutral institutional body and then backs away from the initiative when the institutional body requires neutrality from the sponsor. Legitimacy losses such as these can be even more damaging when membership of different organizations overlaps. This was the case with the ISO, the ECMA, and the Java collective.

The Future of Java

Even as legitimacy problems continued to plague Sun, its activities in the Java product-market intensified. For instance, in 1998, Sun acquired NetDynamics, a vendor of Java-based "enterprise servers," and forged a broad product development alliance with AOL-Netscape. Such a transformation from a relatively benevolent technology sponsor to an active competitor might have been unavoidable as Sun decided to cash in on Java, its most valuable asset. This harvesting strategy, however, could have created a counterproductive perception that Sun's own programmers enjoyed an unfair advantage over other members of the collective in creating Java applications (much as Microsoft's application programmers have been accused of having unfair access to Windows).

As Java matured, other members of the coalition sought to safeguard their own private interests by demanding more control over Java's future. Responding to their pressure, in March 2000, Sun offered to relinquish more of its control over Java (Shankland, 2000b). Sun proposed a new process, called Java Community Process (JCP) 2.0, for the stewardship of the Java standard. JCP 2.0 allows members of the Java coalition to establish working groups to extend the Java standard into new areas and determine when these new specifications will be released to the public. As a spokesperson from Sun stated:

> Java was our little baby. [It was] very hard to let go of its hand when it was taking its first steps. . . . It's almost at the adolescent stage. Maybe we should start letting it walk a bit more on its own. (Shankland, 2000b)

Despite these public announcements and initiatives, new complaints emerged that Sun was unfairly charging partners licensing fees even when they had contributed significantly to the extension of the Java standard under the new community-source licensing initiative (Lyons, 2000). Reflecting on the tension that private interests inevitably generate in initiatives involving collectives, Daniel Lyons, an industry analyst, observed:

> The squabble threatens to rupture the already fractious alliance. IBM and others could start touting a version of Java different from the one Sun pushes. (Lyons, 2000)

It remains to be seen whether Sun manages to walk the fine line between opening up and controlling its Java technology, thereby preventing fragmentation. Whatever the future brings, Sun's sponsorship of the Java standard provides key insights into the nature of the standardization process, the tensions confronted by a technology sponsor as a standard emerges and evolves, and the interactions among strategic actors that together determine the standard's eventual rise or downfall.

DISCUSSION

We began this article by highlighting the important role that institutions play in the functioning of technological fields. We narrowed our focus to examine the emergence of common standards governing the functioning of network technological fields.

In focusing on standards, we began addressing questions of central interest to institutional theorists: How do these standards emerge? What are the challenges associated with sponsoring them? Our objective was to generate insights on the processes associated with institutional entrepreneurship in shaping standards in the making.

Our introductory discussion revealed two paradoxical properties associated with standards. First, standards both enable and constrain. This structurational property of standards makes it difficult for actors to forge agreements that enable activities in the present but have the potential to constrain activities in the future. Second, the creation of standards involves cooperation between competitors. This coopetitional property of standards makes it difficult for actors with private interests to reach consensus on common standards.

Sponsorship Challenges

These structurational and coopetitional properties interact with each other to create a situation wherein a standard is as much a set of rules that firms agree upon as a set of rules that they plan to depart from. Under these conditions, orchestrating the emergence of a standard is a complex task. These difficulties become all the more pronounced when a firm with its own interests, such as Sun, attempts to sponsor its technology as the common standard. There is little doubt that the technology sponsor is attempting to shape emerging institutions for its own competitive gain. It is no wonder that incumbents and newcomers are wary of the sponsor's intentions. Indeed, they carefully assess the sponsor's every move before deciding whether to join or counter these sponsorship efforts.

Mobilization challenges. At the outset, a new standards initiative will have to counter the inertia associated with entrenched standards that enable the functioning of existing technological fields. Additionally, it will encounter resistance from dominant actors in existing technological fields who are threatened by the new standard. As Sun's actions suggest, one way to overcome this initial inertia and resistance is to mobilize a large collective around the new standard. But mobilizing support is not easy. The sponsor has to persuade potential rivals to constrain themselves to a standard that could place them at a competitive disadvantage in the future.

In such instances, a sponsor may attempt to jump-start a collective by placing a portion of its technology in the public domain. This initiative is consistent with an open-systems approach to sponsorship. By opening up its technology, the sponsor may gain the support of firms that wish to gain access to a promising technology without having to invest afresh in a competing alternative. However, this initiative alone may still not be enough to entice a critical mass of firms to join the collective.

The sponsor may also have to galvanize action by evoking images of a collective organized against a powerful and unpopular incumbent. Such an action is consistent with observations in the institutional theory literature on how actors generate momentum by evoking the image of a common enemy (Swaminathan & Wade, 2001). If the sponsor's vision of a redefined technology field free of oppression is alluring enough, even potential rivals may overcome their natural wariness and enlist in the war against the common enemy.

More broadly, these actions illustrate the social skills a sponsor needs to mobilize a collective around its technology. Social skills represent an actor's ability to motivate cooperation in other actors by providing them with common meanings and identities around which actions can be undertaken and justified (Fligstein, 1997; Rao, 1998). By offering its partners a compelling vision of a future technological trajectory, an actor can attract the support of a large number of actors to its emerging collective.

Maintenance challenges. The mobilization of a collective is but the first of several challenges that a sponsor confronts. As the Java case illustrates, members of an emerging collective may begin departing from the emerging standard even as they agree to conform to it. One of the reasons for doing so is political. For instance, an actor can join an emerging collective and attempt to create countermobilization movements from within. Firms may also depart from the standard for strategic reasons. After all, members of a collective may implement the standard in line with their own interpretations of the standard and their needs to differentiate their products in the marketplace.

It makes little difference to the sponsor whether firms are departing for political or strategic reasons. Of greater significance is that such actions can potentially fragment the emerging standard, as members of the collective begin offering multiple versions of the standard. This has been the fate of several other standardization initiatives, most notably Unix.

It is essential to have a uniform implementation of a standard if it is to function effectively. The problem is that, like the sponsor, other firms would also like to have their specifications incorporated into the emerging standard. In this context, if the sponsor is not careful, it may easily lose control over its standard as well as its position in the hierarchy of the emerging technological field. This was the situation that transpired with IBM in the case of personal computers.

Either eventuality—the fragmentation of a standard or the ceding of its control to rivals—is unacceptable to a sponsor. The sponsor can deal with these challenges by establishing specific control mechanisms over the standardization process. For instance, as in the Java case, the sponsor may use legal mechanisms to thwart countermobilization and enforce rigorous testing of different implementations to ensure compatibility.

Clearly, with these initiatives, the sponsor displays a different set of skills from the ones it had used earlier to mobilize the bandwagon. Specifically, the ability and willingness of a sponsor to protect its franchise with legal instruments and enforcement mechanisms represent political skills. Political skills may include strong-arm tactics used to sustain cooperation within a coalition and to keep members with private interests from diverging from the common meanings and identities that the sponsor had established earlier.

Contradictory pressures. When the challenges of mobilizing a coalition are compounded by those of maintaining it, the dilemma of a technology sponsor becomes even starker. Given the sponsor's need to continue with mobilization efforts, the appropriate level of control becomes an important issue for consideration. Too little control may lead to fragmentation or loss of ownership, whereas too much control may stifle the very emergence of the standard.

This dilemma arises because standards evolve even as they are being used to develop compatible products and applications (Garud & Kumaraswamy, 1995b; Jain, 2001). This process reflects the duality implicit in standards in the making, wherein a standard is both the medium and the outcome of action (Giddens, 1984). Under these conditions, orchestrating both the mobilization and the maintenance of the collective is an incredibly complex task for the sponsor.

This task is rendered all the more difficult by the presence of members with private interests within the collective. The process becomes highly politicized as a result. In such a situation, "legitimacy traps" emerge for the sponsor, as it too has private interests at stake. Specifically, as the Java case illustrates, initiatives taken by the sponsor to enforce a relatively uniform meaning structure may easily be perceived by others as attempts to gain control for private gain. Consequently, members of the collective may begin perceiving that the sponsor is not really earnest in its intentions to keep the standard open. It is not surprising, therefore, that Sun's actions to orchestrate compliance with its Java specifications generated doubts about its future intentions.

These observations add to our understanding of how processes associated with standards sponsorship create legitimacy challenges for the sponsor. The structurational and coopetitional properties of standards interact to create legitimacy traps for technology sponsors. Legitimacy traps are especially likely to arise to the extent that rule creation and rule enforcement reside with the sponsor. As rule creator, the sponsor determines which rules are integrated into the standard and when they will be promulgated. As rule enforcer, the sponsor ensures that members of the collective conform to the set of rules it has established for the technology. In taking on both roles, the sponsor requires collective members to conform to current or old rules even as it orchestrates the creation of new rules. In other words, members of the collective often find themselves to be one step behind the sponsor. This may result in a loss of credibility for the sponsor.

Comparison with other Sponsorship Initiatives

It is instructive to compare Sun's sponsorship experience with initiatives in other technological fields. Over the years, there have been many sponsorship initiatives involving an open-systems approach. Examples include IBM's accidental sponsorship of the personal computer field, Apple's efforts to encourage cloning of its Macintosh architecture and, more recently, AOL-Netscape's Mozilla open-source initiative to create a new version of its Navigator Web browser. Firms in technological fields such as consumer electronics have also sponsored their technologies employing an open-systems approach. Popular examples include JVC's sponsorship of its VHS technology in the videocassette recorder field (Hariharan, 1990) and Columbia's sponsorship of its 33 1/3 rpm music-recording technology (Langlois & Robertson, 1992).

However, sponsorship using an open-systems approach does not guarantee success. For instance, AOL-Netscape's Mozilla open-source initiative was plagued by delays that have contributed to Navigator's decline in market share. Moreover, a number of firms have encountered legitimacy traps as part of their respective sponsorship initiatives. IBM's ill-fated attempts at establishing control over the open architecture in the personal computer field by introducing the PS/2 personal computer system with its proprietary microchannel "bus" architecture is one example. Apple too encountered legitimacy problems when it abruptly changed its open-systems strategy midstream by voiding its licensing agreements with clone manufacturers.

Despite these examples of limited success, such sponsorship efforts are increasing in number. Many efforts become intertwined with legitimacy issues as firms claim to be more open than others. This is likely to happen because the structurational property of standards interacts with its coopetitional property. As in the Sun case, initiatives by the sponsor to coordinate a collective around standards are perceived as attempts to establish control for private gain. Specifically, when private interests exist, control mechanisms deployed by the sponsor to coordinate a collective are likely to politicize the open-systems sponsorship model.

In sum, the private interests implicit in coopetition can complicate the already muddy waters created by the structurational property of standards. To appreciate this point, it is useful to compare the Java case with a sponsorship initiative that is currently generating considerable interest—the Linux operating system. A key difference between the two cases is that private interests have so far been muted in the Linux case. As Linux is available free, firms have sought to generate rents by offering support and other services related to the operating system. Therefore, cooperation to create and maintain the standard has not yet been plagued by competition to appropriate benefits. Tensions between rule creation and rule enforcement have largely been kept at bay through a grassroots culture that stresses openness, sharing, and compatibility more than commercial gain (Stallman, 1999). Moreover, challenges associated with the structurational property of standards have been muted because the number of rules associated with the standard have been kept to a minimum, thereby allowing the Linux collective considerable flexibility. Till recently, these facets have led to the rapid growth of the collective around Linux.

Private interests, however, are coming into play in the Linux case as well. Vendors such as Corel and TurboLinux have begun including their own proprietary technologies in the core Linux operating system (Hamm, 1999; Orenstein, 1999). Large established vendors such as IBM have begun endorsing Linux as the "operating system of the Internet" (Lohr, 2000). As private interests generate coopetition, some of the same legitimacy traps that have dogged Sun in the Java case may become apparent in the Linux case too.

Nevertheless, it is possible that private interests are emerging at a stage when Linux has gained sufficient momentum to carry it through the institutionalization process. The timing of the appearance of private interests during a standard's emergence might be a key issue. Private interests, if they appear too early, may dampen a collective's ability to generate sufficient momentum around a standard. This could well be the case with Java.

Summary Insights

We offer summary insights on how the Java case adds to understanding of institutional entrepreneurship through technology sponsorship initiatives within network technological fields. Our first observation pertains to the fragility of the standardization process. The structurational and coopetitional properties of standards interact to create an institutionalization process that is inherently unstable and politicized. Members of a collective may depart from agreements on common standards for political, strategic, or interpretive reasons. This situation is especially true when a sponsor is unable or unwilling to deliver upon promises in a timely manner, a situation that transpired in Sun's case. At the margin, it may be difficult to ascertain whether a member of the collective is extending or fragmenting the emerging standard.

Even as firms begin diverging from a common standard, they want their own innovations to be built into it. A fresh round of negotiations ensues as firms bargain with one another to determine the composition of the evolving standard. Moreover, as the number of firms that constitute the standards collective increases, the challenges associated with coordinating the collective also increase. At this point, firms may decide to leave the collective and offer their own versions of the standard. With time, these coevolutionary dynamics, if left unattended, can fragment the emerging standard. In sum, institutionalization processes involving standards' creation and maintenance are inherently fragile.

Under these conditions, it is difficult even for a neutral body to sponsor a common standard. These challenges are all the more pronounced when a firm seeks to sponsor a common standard. Specifically, a sponsor's private interests begin generating credibility problems when it begins enforcing rules that it helped to create. Often, sponsors do not address these problems ex ante as they become clear only once the constraining facets of standards become evident to collective members. When the constraints become clearer, members may attempt to depart from the collective.

To orchestrate the sponsorship effort, the sponsor needs to deploy both social and political skills. Whereas social skills represent an actor's ability to motivate cooperation in other actors by providing them with common meanings and identities, political skills represent its ability to sustain cooperation when private interests force divergence from these common meanings and identities. A sponsor has to exercise both these skills simultaneously as it tries to mobilize a collective, even as it attempts to coordinate its activities. The challenge in exercising both skills become apparent when one understands that these skills themselves contrast with one another. In this context, strategic action is an attempt by the sponsor to create and sustain a semblance of stability given the fragility of the standardization process.

These insights, one at the level of standards and the other at the level of the technology sponsor, were well captured by Sun's vice president, Jon Kannegaard, who likened his company's stewardship of Java to a car "that has two wheels on the ground and two wheels off and is always about to tip over" (Gage, 1999).

CONCLUSION

This study offers us several insights on institutional entrepreneurship in emerging technological fields. Our observations suggest that common technological standards, which are a key facet of the institutional environment of network technological fields, have built-in tensions. All standards have enabling and constraining effects, and they are often forged through cooperation among competitors. This process generates temporary, partial agreements by interdependent parties with private and sometimes diverging interests. The challenge of engendering collective action is further complicated by intertemporal inconsistencies between the initiatives required to mobilize a collective and those required to maintain it. These tensions make it difficult for a firm to sponsor its proprietary technology as a common standard.

These observations suggest the need to closely examine standards in the making in order to understand the institutionalization of technological fields. The Java case illustrates how the standardization process can be co-opted by means of impression management, sense making, attempts to change the rules of the game by appeal to authority, competing sources of legitimacy, and loose coupling between institutions and their sponsors. The creation of institutions, even technical ones, is messy, manipulative, instrumental, conscious, and devious. Essentially, even the development of technological standards is a battle fought in political and cognitive realms.[4]

An appreciation of these processes offers us an opportunity to reflect on the nature and scope of the agency involved in sponsoring common standards. A sponsor walks a fine line in creating a common standard that enables rather than constrains the emergence of a technological field. At any stage of the sponsorship effort, a sponsor has to maintain a common standard even as it is being created. Maintenance and creation require political and social skills, respectively. As one might expect, these skills themselves contradict each other. Institutional entrepreneurship, then, may require an ability to deploy these skills in such a manner that they do not exacerbate the contradictions that are inherent in standards' creation and maintenance but, instead, provide a synthesis.

Our theoretical frame can be gainfully employed to study the interactions between the micro actions of actors and emerging macro institutional structures (Fligstein, 1999; Hirsch & Lounsbury, 1997; Schelling, 1978) in other settings. There are many cases in the contemporary realm that qualify as relevant strategic research sites. These include mobile telephony, broadband communications, digital music distribution, and data communications. One line of inquiry could focus on the timing and extent to which institutional entrepreneurs open up or control their technologies. Another line of inquiry could explore the kinds of governance mechanisms that institutional entrepreneurs put in place to coordinate and regulate the activities of a collective. Such inquiry would help in developing a body of knowledge about institutional entrepreneurship in emerging technological fields.

[4] We are indebted to an anonymous reviewer for this input.

REFERENCES

Aldrich, H. E., & Fiol, C. M. 1994. Fools rush in? The institutional context of industry creation. ***Academy of Management Review,*** 19: 645–670.

Arthur, B. 1989. Competing technologies, increasing returns and lock-in by historically small events. ***Economic Journal,*** 99: 116–131.

Bank, D. 1995. The Java saga. ***Wired News*** *(www.wired.com/wired/archive/3.12/java.saga.html):* December 3.

Barnett, W. 1990. The organizational ecology of a technological system. ***Administrative Science Quarterly,*** 35: 31–60.

Bijker, W. B., Hughes, T. P., & Pinch, T. 1987. ***The social construction of technological systems.*** Cambridge, MA: MIT Press.

Bingley, L. 1999. Sun blames Microsoft as Java ISO plans die. ***PCWeek Online*** *(www.zdnet.com/zdnn/stories/news/0,4586,1014537,00.html):* April 29.

Brandenburger, A., & Nalebuff, B. J. 1995. The right game:

Use game theory to shape strategy. ***Harvard Business Review,*** 76(3): 57–71.

Brunsson, N. S., & Jacobsson, B. 2000. ***A world of standards.*** Oxford, England: Oxford University Press.

Callon, M. 1986. The case of the electric vehicle. In M. Callon, D. Law, & A. Rip (Eds.), ***Mapping the dynamics of science and technology:*** 19–34. London: MacMillan.

Christensen, S., Karnøe, P., Pedersen, J. S., & Dobbin, F. 1997. Actors and institutions. ***American Behavioral Scientist,*** 40: 392–396.

Clark, K. 1985. The interaction of design hierarchies and market concepts in technological evolution. ***Research Policy,*** 14: 235–251.

Clark, T. 1998. Sun shakes up Java groups. ***CNET News.com*** *(news.cnet.com/category/0-1003-200-329645.html):* May 27.

Computerworld. 1998. Call it Java—For a price. December 7: 90.

Constant, E. 1980. ***The origins of the turbojet revolution.*** Baltimore: Johns Hopkins University Press.

Dacin, M. T., Ventresca, M. J., & Beale, B. D. 1999. The embeddedness of organizations: Dialogue and directions. ***Journal of Management,*** 25: 317–356.

Darrow, B. 2000. Sun, panel work on Java community process. ***Techweb.com*** *(www.techweb.com/wire/story/TWB20000412S0001):* April 12.

David, P. A. 1985. Clio and the economics of QWERTY. ***American Economic Review.*** 75: 332–337.

DiMaggio, P. 1988. Interest and agency in institutional theory. In L. Zucker (Ed.), ***Institutional patterns and culture:*** 3–22. Cambridge, MA: Ballinger.

Dobbin, F., & Dowd, T. 1997. How policy shapes competition: Early railroad foundings in Massachusetts. ***Administrative Science Quarterly,*** 42: 501–529.

Dosi, G. 1982. Technological paradigms and technological trajectories. ***Research Policy,*** 11: 147–162.

Dow Jones Newswires. 1997. High-tech group asks Sun to give up control of Java (WSJ interactive version). September 11.

Elmer-Dewitt, P. 1996. Why Java is hot. ***Time,*** January 22: 58–60.

Farrell, J., & Saloner, G. 1986. Installed base and compatibility: Innovation, product preannouncements and predation. ***American Economic Review,*** 76: 940–955.

Ferguson, C. H., & Morris, C. R. 1993. ***Computer wars.*** New York: Random House.

Fligstein, N. 1997. Social skill and institutional theory. ***American Behavioral Scientist,*** 40: 397–405.

Fligstein, N. 1999. ***Social skill and the theory of fields.*** Working paper, University of California, Berkeley.

Gage, D. 1997. Sun moves Java standardization forward. ***Smart Reseller Online*** *(www.techweb.com/wire/news/1997/09/0922standard.html):* September 22.

Gage, D. 1999. Who really owns Java? ***Smart Reseller Online*** *(www.zdnet.com/sr/stories/news/0,4538,2403818-6,00.html):* December 5.

Garud, R., & Jain, S. 1996. The embeddedness of technological systems. In J. Baum & J. Dutton (Eds.), ***Advances in strategic management,*** vol. 13: 389–408. Greenwich, CT: JAI Press.

Garud, R., Jain S., & Phelps, C. 1998. Technological linkages & transience in network fields: New competitive realities. In J. Baum (Ed.), ***Advances in strategic management,*** vol. 14: 205–237. Greenwich, CT: JAI Press.

Garud, R., & Karnøe, P. 2002. Bricolage vs. breakthrough: Distributed and embedded agency in technology entrepreneurship. ***Research Policy:*** In press.

Garud, R., & Kumaraswamy, A. 1993. Changing competitive dynamics in network industries: An exploration of Sun Microsystems' open systems strategy. ***Strategic Management Journal,*** 14: 351–369.

Garud, R., & Kumaraswamy, A. 1995a. Technological and organizational designs to achieve economies of substitution. ***Strategic Management Journal,*** 16: 93–110.

Garud, R., & Kumaraswamy, A. 1995b. Coupling the technical and institutional faces of Janus in network industries. In R. Scott & S. Christensen (Eds.), ***The institutional construction of organizations:*** 226–242. New York: Sage.

Garud, R., & Rappa, M. 1994. A socio-cognitive model of technology evolution. ***Organization Science,*** 5: 344–362.

Giddens, A. 1979. ***Central problems in social theory.*** Berkeley: University of California Press.

Giddens, A. 1984. ***The constitution of society: Outline of the theory of structuration.*** Berkeley: University of California Press.

Gilmore, D. 1998. Sun could learn a lesson on Java from Netscape. ***Mercury News Online,*** September 18.

Glaser, B., & Strauss, A. 1967. ***The discovery of grounded theory: Strategies of qualitative research.*** London: Wiedenfeld & Nicholson.

Gomes, L. 1998a. Hewlett-Packard to market its own strain of Java. ***Wall Street Journal,*** March 20: A3.

Gomes, L. 1998b. Sun loosening its control over Java language. ***Wall Street Journal,*** December 8: B6.

Gould, S. J. 1980. ***The panda's thumb.*** New York: Viking Penguin.

Granovetter, M. 1985. Economic action and social structure: the problem of embeddedness. ***American Journal of Sociology,*** 91: 481–510.

Hamel, G., & Prahalad, C. K. 1994. ***Competing for the future.*** Boston: Harvard Business School Press.

Hamm, S. 1999. The wild and woolly world of Linux. ***BusinessWeek,*** November 15: 130–134.

Hardin, R. 1982. ***Collective action.*** Baltimore: Johns Hopkins University Press.

Hariharan, S. 1990. ***Technology compatibility, standards and global competition: The dynamics of industry evolution and competitive strategies,*** Unpublished doctoral dissertation, University of Michigan, Ann Arbor.

Helft, M., & Mardesich, J. 1997. Sun files suit to guard Java from Microsoft. ***Mercury News Online,*** October 8.

Helft, M., & Quinlan, T. 1998. E-mails show animosity over Java. ***Mercury News Online,*** November 5.

Henderson, R. M., & Clark, K. 1990. Architectural innovation: The reconfiguration of existing product technologies and the failure of established firms. ***Administrative Science Quarterly,*** 35: 9–30.

Hirsch, P. M. 1975. Organizational effectiveness and the institutional environment. ***Administrative Science Quarterly,*** 20: 327–344.

Hirsch, P. M., & Lounsbury, M. 1997. Ending the family quarrel. ***American Behavioral Scientist,*** 40: 406–418.

Hughes, T. 1983. ***Networks of power.*** Baltimore: Johns Hopkins University Press.

Jain, S. 2001. ***A process framework of collective standards emergence.*** Unpublished doctoral dissertation, New York University.

Jepperson, R. L. 1991. Institutions, institutional effects and institutionalism. In W. W. Powell & P. J. DiMaggio (Eds.), ***The new institutionalism in organizational analysis:*** 143–163. Chicago: University of Chicago Press.

Jick, T. 1979. Mixing qualitative and quantitative methods: Triangulation in action. ***Administrative Science Quarterly,*** 24: 602–611.

Jones, C. 1997. Sun's not bowing in standards battle. ***Wired News*** *(www.wired.com/news/news/technology/story/7101.html):* September 22.

Karnøe, P. 1999. The business systems framework and the Danish SME's. In P. Karnøe, P. H. Kristensen, & P. H. Andersen (Eds.), ***Mobilizing resources and generating competencies—The remarkable success of small and medium sized enterprises in the Danish business system:*** 7–72. Copenhagen: Copenhagen Business School Press.

Katz, M. L., & Shapiro, C. 1985. Network externalities, competition, and compatibility. ***American Economic Review,*** 75: 424–440.

Kenney, M. 2000. ***Silicon Valley: Anatomy of an entrepreneurial region.*** Stanford University Press.

Kirsner, S. 1997. Coke, Nike . . . Java? A battle for brand. ***Wired News*** *(www.wired.com/news/news/business/java/story/8904.html):* December 8.

Langlois, R. N., & Robertson, P. L. 1992. Networks and innovation in a modular system: Lessons from the microcomputer and stereo component industries. ***Research Policy,*** 21: 297–313.

Lincoln, Y. S., & Guba, E. G. 1985. ***Naturalistic inquiry.*** New York: Sage.

Lohr, S. 2000. A mainstream giant goes countercultural. ***New York Times,*** March 20: C1.

Lukes, S. 1974. ***Power: A radical view.*** London: Macmillan.

Lyons, D. 2000. Bitter brew. ***Forbes.*** February 21.

Meyer, J. W., & Rowan, B. 1977. Institutionalized organizations: Formal structure as myth and ceremony. ***American Journal of Sociology,*** 83: 340–363.

Miles, M., & Huberman, M. A. 1984. ***Qualitative data analysis: A source book of new methods.*** Beverly Hills, CA: Sage.

Moeller, M. 1997. Sun's product plans alarm Java licensees. ***PC Week Online*** *(www.zdnet.com/pcweek/news/1215/15java.html):* December 15.

Mohr, L. B. 1982. ***Explaining organizational behavior: The limits and possibilities of theory and research.*** San Francisco: Jossey-Bass.

Nerney, C. 1997. JavaSoft fails to follow roadmap. ***Network World Online*** *(www.nwfusion.com/news/1997/1208java.html):* December 8.

Nerney, C. 1998. Sun's handling of Java criticized. ***Network World Online*** *(www.nwfusion.com/news/0608java.html):* June 8.

North, D. C. 1990. ***Institutions, institutional change and economic performance.*** New York: Cambridge University Press.

Oakes, C. 1998. Policing the Java police. ***Wired News,*** *(www.wired.com/news/news/technology/story/13685.html):* July 14.

Olson, M. 1965. ***The logic of collective action: Public goods and the theory of groups.*** Cambridge, MA: Harvard University Press.

Orenstein, D. 1999. Linux server cluster threatens open unity. ***Computerworld,*** November 1: 12.

Pinch, T. J., & Bijker, W. E. 1987. The social construction of facts and artifacts: Or how the sociology of science and the sociology of technology might benefit each other. In W. E. Bijker, T. P. Hughes, & T. J. Pinch (Eds.), ***The social construction of technological systems:*** 17–50. Cambridge, MA: MIT Press.

Porac, J., Rosa, J., Spanjol, J., & Saxon, M. 2001. America's family vehicle: Path creation in the U.S. minivan market. In R. Garud & P. Karnøe (Eds.), ***Path dependence and creation:*** 213–242.

Powell, W. W., & DiMaggio, P. J. 1991. ***The new institutionalism in organizational analysis.*** Chicago: University of Chicago Press.

Rao, H. 1994. The social construction of reputation: Certification contests, legitimation and the survival of organizations in the American automobile industry, 1895–1912. ***Strategic Management Journal.*** 15: 29–44.

Rao, H. 1998. Caveat emptor: The construction of non-profit consumer watchdog organizations. ***American Journal of Sociology,*** 103: 912–961.

Raymond, E. 1999. ***The magic cauldron*** *(www.tuxedo.org/~esr/writings/magic-cauldron/magic-cauldron.html):* Accessed April 16.

Rein, L. 1997. Microsoft pushes Java aside. ***Wired News*** *(www.wired.com/news/news/technology/story/7324.html):* September 30.

Rosenberg, N. 1982. ***Inside the black box: Technology and economics.*** Cambridge, England: Cambridge University Press.

Rosenkopf, L., & Tushman, M. L. 1994. The coevolution of technology and organization. In J. Baum & J. Singh (Eds.), ***Evolutionary dynamics of organizations:*** 403–424. New York: Oxford University Press.

Sanchez, R. 1995. Strategic flexibility in product competition. ***Strategic Management Journal,*** 16: 135–159.

Schelling, T. C. 1978. ***Macromotives and microbehavior.*** New York: Norton.

Schilling, M. A. 2000. Towards a general modular systems theory and its application to interfirm product modularity. ***Academy of Management Review,*** 25: 312–334.

Schlender, B. 1997. The adventures of Scott McNealy: Javaman. ***Fortune,*** October 13: 70–78.

Schumpeter, J. 1942. ***Capitalism, socialism and democracy.*** New York: Harper & Brothers.

Scott, W. R. 1994. Institutional analysis: Variance and process theory approaches. In W. R. Scott, J. W. Meyer, and associates (Eds.), ***Institutional environments and organizations: Structural complexity and individualism:*** 81–99. Thousand Oaks, CA: Sage.

Scott, W. R. 1995. ***Institutions and organizations.*** Thousand Oaks, CA: Sage.

Scott, W. R., & Christensen, S. 1995. ***The institutional construction of organizations.*** Thousand Oaks, CA: Sage.

Selznick, P. 1957. ***Leadership in administration.*** New York: Harper & Row.

Shankland, S. 2000a. Sun adjusting terms of Java server software licensing. ***CNET News.com*** *(news.cnet.com/news/0-1003-200-1616607.html).* March 31.

Shankland, S. 2000b. Sun offers olive branch to Java partners. ***CNET News.com*** *(news.cnet.com/news/0-1003-200-1562301.html):* March 1.

Shapiro, C., & Varian, H. R. 1999. The art of standards wars. ***California Management Review,*** 41(2): 9–32.

Sliwa, C. 1998. Java standards process explained. ***Computerworld,*** October 26: 8.

Sliwa, C. 1999a. Standards body nixes alternate Java plan. ***Computerworld,*** January 15.

Sliwa, C. 1999b. Sun drops Java standard, ***Computerworld,*** December 13: 1.

Stallman, R. M. 1999. ***The GNU project*** *(www.gnu.org/gnu/thegnuproject.html):* Accessed April 28.

Suchman, M. C. 1994. ***On advice of counsel: Law firms and venture capital funds as information intermediaries in the structuration of Silicon Valley.*** Unpublished doctoral dissertation, Stanford University, Stanford, CA.

Sun Microsystems. 1999. ***Java: The first 800 days*** *(java.sun.com/events/jibe/timeline.html):* Accessed April 16.

Surowiecki, J. 1997. Culture wars: Behind the Java hype is a struggle for computer-world hegemony. ***Slate*** *(http://slate.msn.com/?id-2621):* September 27.

Swaminathan, A., & Wade, J. 2001. Social movement theory and the evolution of new organizational forms. In C. B. Schoonhoven & E. Romanelli (Eds.), ***The entrepreneurship dynamic in population evolution:*** 286–313. Stanford, CA: Stanford University Press.

Tsoukas, H. 1989. The epistemological status of idiographic research in the comparative study of organizations: A realist perspective. ***Academy of Management Review,*** 14: 551–561.

Tushman, M. L., & Anderson, P. 1986. Technological discontinuities and organizational environments. ***Administrative Science Quarterly,*** 31: 439–465.

Utterback, J., & Abernathy, W. 1975. A dynamic model of process and product innovation. ***Omega,*** 33: 639–656.

Uzzi, B. 1996. The sources and consequences of embeddedness for the economic performance of organizations: The network effect. ***American Sociological Review,*** 61: 674–698.

Van den Belt, H., & Rip, A. 1987. The Nelson-Winter-Dosi model and synthetic dye chemistry. In W. Bijker, T. Hughes & T. Pinch (Eds.), ***The social construction of technological systems:*** 135–158. Cambridge, MA: MIT Press.

Van de Ven, A. H., & Garud, R. 1989. A framework for understanding the emergence of new industries. In R. Rosenbloom & R. Burgelman (Eds.), ***Research on technological innovation and management policy,*** vol. 4: 195–226. Greenwich, CT: JAI Press.

Van de Ven, A. H., & Garud, R. 1994. The coevolution of technical and institutional events in the development of an innovation. In J. Baum & J. Singh (Eds.), ***Evolutionary dynamics of organizations:*** 425–443. New York: Oxford University Press.

Wade, J. 1995. Dynamics of organizational communities and technological bandwagons: An empirical investigation of community evolution in the microprocessor market. ***Strategic Management Journal***, 16: 111–133.

Wade, J. 1996. A community-level analysis of sources and rates of technological variation in the microprocessor model. ***Academy of Management Journal***, 39: 1218–1244.

Weick, K. 1979. ***The social psychology of organizing***. New York: Random House.

Wingfield, N. 1995. All-out Internet war begins: Microsoft enters fray. ***Infoworld***, December 4: 1.

Wolfe, A. 1998. Sun, HP square off over real-time Java spec. ***EE Times Online*** *(www.eetimes.com/story/OEG19981030S0015)*: November 2.

Raghu Garud *(rgarud@stern.nyu.edu)* is an associate professor of management and organizational behavior at the Leonard N. Stern School of Business, New York University. Professor Garud's interests lie at the nexus of technology, strategy, and organizations. His publications explore path creation, metamorphic change, new organizational forms, economies of substitution, researcher persistence, and technology embeddedness.

Sanjay Jain *(sjain@bus.wisc.edu)* is an assistant professor of management and human resources and an associate of the Weinert Center for Entrepreneurship in the School of Business at University of Wisconsin—Madison. He received his Ph.D. from the Stern School of Business at New York University. His current research focuses on examining the strategic and organizational issues associated with technology standards as well as understanding the competitive and cooperative dynamics that unfold in network-based industries.

Arun Kumaraswamy *(akumaras@crab.rutgers.edu)* is an assistant professor of management in the School of Business at Rutgers, The State University of New Jersey. Professor Kumaraswamy's primary research interest is the application of a real options perspective to investments in and the organization of R&D. His other interests include competitive dynamics and technology evolution—especially the sponsorship of technological standards—in high-technology industries.