December 2004

Community Learning in Information Technology Fashion

Ping Wang  
*University of California, Los Angeles*

Neil Ramiller  
*Portland State University*

Follow this and additional works at: [http://aisel.aisnet.org/icis2004](http://aisel.aisnet.org/icis2004)

Recommended Citation
[http://aisel.aisnet.org/icis2004/4](http://aisel.aisnet.org/icis2004/4)

This material is brought to you by the International Conference on Information Systems (ICIS) at AIS Electronic Library (AISeL). It has been accepted for inclusion in ICIS 2004 Proceedings by an authorized administrator of AIS Electronic Library (AISeL). For more information, please contact elibrary@aisnet.org.
COMMUNITY LEARNING IN INFORMATION TECHNOLOGY FASHION

Ping Wang
Anderson School of Management
University of California, Los Angeles
Los Angeles, CA  U.S.A.
pwang@anderson.ucla.edu

Neil C. Ramiller
School of Business Administration
Portland State University
Portland, OR  U.S.A.
neilr@sb.apdx.edu

Abstract

In their striving to learn about information technology innovations, organizations draw on knowledge resources available in the diverse organizational communities that converge around those innovations. But even as such organizations learn about an innovation, so too does the larger community. Community learning takes place as its members reflect upon their learning and contribute their experiences, observations, and insights to the community's on-going innovation discourse. Community learning and organizational learning thus build upon one another in a reciprocal process, or cycle, over time, as the stock of interpretations, adoption rationales, implementation strategies, and utilization patterns is expanded and refined. Relative to this overall cycle, we explore the neglected aspect that concerns how community learning draws on organizational learning. Analyzing the community discourse on enterprise resource planning (ERP) over the past 14 years, we found that different types of organizational actors played different roles, at different times, in contributing different types of knowledge to the discourse. Research analysts and technology vendors took leadership early on in articulating the know-what (conceptualization and interpretation) and know-why (justification) for ERP. Later on, adopters came to dominate the discourse through contributions of know-how (capacitation). We situate these observations in a larger model of the learning cycle, and outline a number of areas for future research on the crucial interactions between organizational and community learning in IT innovation.

Keywords: Information technology innovation, organizational community, community learning, organizational learning, knowledge, discourse, information technology fashion

Introduction

Information Systems researchers have recently shown increasing interest in organizational learning. A number of studies find that organizational learning plays important roles in, for example, lowering the barriers to innovation with information technologies (Fichman and Kemerer 1997), overcoming difficulties in implementing IT (Ang et al. 1997), and enhancing firm performance with IT (Tippins and Sohi 2003). In most studies, however, organizational learning is portrayed as a learning-by-doing process that begins when a firm adopts and implements the new IT. We want to point out that firms also do something we will call learning about the technology.

Learning-about and learning-by-doing are conceptually different processes. In learning by doing an IT, a learner creates knowledge by developing, producing, implementing, or operating the technology. In learning about an IT, a learner creates knowledge by making sense of information that is available apart from direct engagement with the technology. Consider Web services, for example. By reading articles and books and attending conferences, one can learn a lot, for example, about the enabling standards and protocols, basic architecture, potential applications, and implementation issues. Of course, to become a developer, vendor, or user of Web services, one will have to go further and learn by doing Web services, for example, developing code for, producing, or implementing and using the services.
While there is now substantial IS research focusing on organizational learning-by-doing, few studies address the learning-about process. Nevertheless, more scholarly attention to learning-about appears warranted, for two reasons. First, learning about IT is often difficult. Prospective adopters struggle over a variety of issues: Which IT to invest in learning about? Why learn about it? Where and from whom to learn? How to evaluate what is learned? Second, what firms learn (or fail to learn) about the IT helps shape the conditions for their learning-by-doing. For example, it is apparent that the preparedness of many firms’ for implementing enterprise resource planning systems (ERP) would have benefited from learning more, up-front, about the substantial difficulty and business impact of this innovation (Davenport 1998).

Learning-about is even more pressing, and problematic, when the IT innovation in question becomes a fashion among firms (Abrahamson 1996). In the past several decades, many IT innovations have undergone extreme swings in popularity. When an IT innovation is widely believed to be at the cutting-edge, firms may be prone to “take a leap of faith” into it without learning enough about it (Slater 1995). Lacking a sufficient understanding going in, such firms may later find the IT to be useless or dysfunctional, or discover that it requires special, hard-to-find skills to implement. Hence, for want of learning-about, implementation of an innovation may be delayed (Fichman and Kemerer 1999) or fail totally (May 2003), contributing to the billions of dollars spent on unused or abandoned information technologies every year (source: Gartner Dataquest).

Except for the inventors of the technologies, all firms learn about new IT from outside sources such as peer firms, trading partners, consultancies, trade journals, and universities. This multiplicity of knowledge sources speaks to the fact that each IT innovation is associated with heterogeneous and interdependent interests that are brought together as an organizational community (Swanson and Ramiller 1997). In this community, some talk and write about the IT; others learn about the IT by listening and reading; still others do both. In reciprocal fashion, then, as organizations learn about an IT innovation from knowledge sources in the larger community, their reflection on that learning feeds back into the knowledge base of the community. Hence, the community learns as its members learn, and vice versa, in a cycle that builds knowledge on a multilevel basis over time.

We argue, then, that an important step in understanding how organizations learn about IT innovations is to study the community learning that both supports organizational learning and feeds upon it. We report on a study of such community learning here. Analyzing the discourse on ERP over the past 14 years, we found that a diverse interorganizational community learned about ERP by incorporating three types of knowledge across a multi-period process:

- **Know-what**: Interpretations that helped to conceptualize ERP.
- **Know-why**: Rationales for adoptions that facilitated the justification of ERP.
- **Know-how**: Implementation and utilization strategies and capabilities that capacitated ERP.

We observed, diachronically, that research analysts’ and vendors’ leadership in conceptualizing and justifying ERP was later replaced by technology adopters’ contributions concerning capacitation of this innovation.

Our study broadens the IT learning literature along two dimensions: toward learning-about, as a complement of learning-by-doing, and toward community learning, as an extension of the traditional attention to individual, group, and organizational learning. The findings illuminate selected aspects of the cycle linking organizational and community learning, specifically unveiling patterns in the constitution of organizational contributions to community learning. The current study also helps point the way toward further study of other aspects of this cycle.

The next section briefly reviews the relevant literature on learning. Then the notion of community learning is further amplified. After describing our research methods, we examine our findings concerning the patterns that are evident in community learning about ERP.

**Learning: Some Key Distinctions**

Our purpose here is to highlight the two gaps we noted above, concerning learning-about and learning by organizational communities. As we remarked, management research on learning commonly takes a learning-by-doing perspective. However, when

---

1For comprehensive reviews of organizational learning research, see Cohen and Sproull (1995). For a review of learning research in IS, see Robey et al. (2000).
Our distinction between learning-by-doing and learning-about is based on Bruner’s (1996) distinction between learning-to-be and learning-about. In the education context, one can learn about many professions, but learning to be a member of a profession requires the learner to do what members of that profession do.

Not only can organizations learn without doing; under certain conditions they must. Confronted with a plethora of IT innovations, a firm simply cannot learn by doing all of them. Moreover, the firm often has to learn about a new IT before deciding to venture into learning by doing it. Absent its own experience with the IT, then, a firm must learn through engagement with others’ ideas, claims, and experiences. The firm, accordingly, learns as part of a larger collective—a fact that brings us next to the issue of collective learning.

Two approaches have been employed to study the collective nature of learning: (1) learning in a collective, and (2) learning by a collective. Under learning in a collective, the learner belongs to an epistemic collective (e.g., a working group, organization, industry, profession, nation-state, etc.). Common knowledge existing in the collective enables interactions among the learners, and it is through these interactions that each learner creates new knowledge. Under learning by a collective, as the learners learn, they encode their experiences and knowledge into artifacts possessed by the collective, including text documents, other media, software, and replicable routines. The collective itself, then, can be said to learn.

In the context of our interest in learning by a collective, the learners that constitute the collective are organizations. In what follows, then, we attempt to amplify our case for paying greater attention to learning by a special form collective—organizational community. Moreover, we argue that, for IT innovations, organizational learning-about and learning by the organizational community are integrally related. We begin by elaborating on the notion of community learning in the context of IT fashions.

Community Learning in IT Fashion

The organizational community we refer to in this paper is a set of organizations with interests in the development and utilization of a specific IT innovation. Such a community comprises a dynamically evolving collection of members from various industries that share a focus on the innovation but are differentiated by the particular interests that motivate them. Interests, indeed, are diverse: vendors want to sell their products; consultants want to sell their services; journalists and editors want readers and advertisers for their magazines; prospective adopters want to make sense of the innovation and, ultimately, sound choices concerning its deployment; and so on. While diverse, these interests are also interdependent. For one thing, many of the community’s members are materially interdependent, participating in a mutually reliant value network made up of suppliers, intermediaries, and customers. Secondly, there is also a kind of interpretive interdependence, because each member’s understanding of the IT is dependent on and subject to the “cycles of interpretation” taking place in the larger community (Swanson and Ramiller 1997). This interpretive interdependence points to learning as a community undertaking.

We define community learning as the process by which an organizational community encodes knowledge into its discourse. How does such community learning happen? At least two recent research streams speak to this question. First, reflecting on the constitution and diffusion of IT innovations, Swanson and Ramiller (1997) propose that a diverse interorganizational community creates and employs an organizing vision, which they define as a collective view for applying new IT in organizations. The organizing vision helps facilitate the innovation’s diffusion by interpreting and legitimating it, and mobilizing associated material resources. The organizing vision is a discursive construction: The community produces and sustains the organizing vision through its on-going conversation. Although organizing vision theory does not address the particulars of the community’s learning process, an organizing vision, as a set of ideas created and modified collectively by members of a community, clearly entails learning. Reciprocally, for any organization wanting to learn about an IT innovation, the organizing vision offers, in broad strokes, collectively constructed knowledge about benefits, costs, and implementation approaches.

---

3 Our distinction between learning-by-doing and learning-about is based on Bruner’s (1996) distinction between learning-to-be and learning-about. In the education context, one can learn about many professions, but learning to be a member of a profession requires the learner to do what members of that profession do.

4 There exist research efforts that explore learning by other types of collectives such as groups or communities of individuals (e.g., Lave and Wenger 1991), organizational populations (e.g., Miner and Haunschild 1995), and geographic regions (e.g., Asheim 1996).

5 The organizational community concept has its origins in ecological theory (e.g., Astley 1985).
Second, scholars studying the diffusion of popular management techniques have recently developed management fashion theory. Abrahamson and Fairchild (1999) define a management fashion as a relatively transitory collective belief, “disseminated by the discourse of management-knowledge entrepreneurs, that a management technique is at the forefront of rational management progress” (p. 709). Management fashion theorists argue that management knowledge entrepreneurs (e.g., consultants, journalist, scholars), competing in a market for ideas, sense managers’ collective demand for new management techniques and produce discourse to promote certain techniques (Abrahamson 1996; Abrahamson and Fairchild 2001). Although a management knowledge market is somewhat different from a community in our sense, some of the insights of management fashion theory are helpful in understanding community learning. For example, the observation that knowledge entrepreneurs from different industries (e.g., consulting, media, business education) differ in when and how fast they spread management ideas is helpful in understanding the differentiated roles that members play in an IT community.

Moreover, IT innovations themselves have fashionable aspects. In the past several decades, many IT innovations have undergone wide swings in popularity. Expert systems, CASE, client-server computing, and, more recently, ERP and CRM constitute notable examples. Although management fashion theorists are primarily concerned with administrative techniques, their definition of fashion is broadly applicable to IT innovations such as these. Where the community discourse associated with an IT innovation reaches a level that sustains a transitory collective belief that the innovation is at the cutting-edge, it becomes a fashion of the type that Abrahamson and colleagues describe.6

IT fashions pose an acute challenge to community learning. When an IT innovation comes into fashion, many firms may mindlessly “jump on the bandwagon” by adopting it with at best a superficial understanding of it (Swanson and Ramiller 2004). Anecdotal evidence shows that such firms sometimes find the adopted IT useless; in other cases they find it extremely difficult to implement and even abandon it. Such lack of understanding suggests that, while some of the trouble can be traced to the foolish behavior of individual firms, problems can also arise where community learning fails to generate the knowledge needed (at least in a timely way) for firms to adequately learn about the new IT.

The exaggerated conditions presented by IT fashions thus highlight the importance of community learning, more generally, to the innovative processes of individual organizations. This, then, returns us to the motivating issue in the study we report here. If organizations learn about IT innovations by drawing on knowledge created and diffused within an interfirm, multi-industry community, then more study needs to be made of the processes and implications of community learning.

Our central research questions, accordingly, are: How does an organizational community encode knowledge about an IT innovation into its discourse? More specifically, who contributes the knowledge that gets encoded in the community discourse? What knowledge is encoded? How do the knowledge and the community producing it evolve as the IT innovation comes into and then goes out of fashion?

Methods

We chose to study the community learning associated with the fashion for ERP. ERP made a good subject for two reasons. First, the organizational community for ERP has been large, diverse, and highly productive of discourse. Since ERP was introduced in 1990, some organizations have been directly involved in the design, production, implementation, and utilization of ERP systems (e.g., vendors, consultants, and adopters); others have been keen to study, evaluate, and analyze the ERP concept, products, services, and market (e.g., research analysts, journalists, and academics). Together, the community they formed based on their diverse but interdependent interests generated a substantial amount of discourse. Second, ERP offers an interesting tension, and conflicting evidence, when it comes to the accomplishment of bonafide community learning. On the one hand, consultants and the trade press have been criticized for fanning the flames of ERP (Slater 1995), while companies have been blamed for imprudently jumping on the bandwagon without understanding the cost and management challenges (Davenport 1998). On the other hand, there has also been much talk of “lessons learned” from implementation and utilization (Keller 1999).

---

5For example, the management knowledge market is conceptualized as a market for knowledge about all kinds of management techniques. In contrast, a community in our sense addresses one particular IT innovation. Of course, the memberships of different innovation communities may overlap.

6It is important to note that coming into fashion does not imply that the innovation has no practical merit.
A community learns by encoding knowledge into its discourse, a process that leaves behind a trail of discursive artifacts that record instances of knowledge and that often identify their contributors. In this study, we relied on this discursive trail to observe the community learning process. Specifically, we selected articles on ERP from ABI/Inform (ABI hereafter) according to three criteria (Saunders et al. 2003). First, the term enterprise resource planning had to appear in the title, abstract, or subject of the article. Second, each selected article had to address primarily the adoption, implementation, or utilization of ERP in organizations. Third, we selected articles only from practitioners’ publications. Although we academics can make significant contributions to the community discourse, most of our contributions to the broader community are delivered through discourse vehicles (classroom lectures, conference presentations, books, and the like) other than academic periodicals, which are largely read by our peers (Benbasat and Zmud 1999). Among the 233 articles meeting the above criteria, we randomly selected 100 for further analysis. Figure 1 indicates that our selection is representative of the overall trend in the volume of ERP discourse captured by ABI.

To identify instances of knowledge embedded in the discourse, we employed a typology that distinguishes between the three components of knowledge noted earlier: know-how, know-why, and know-what. Such a typology has been applied in the technology innovation literature (e.g., Lundvall and Johnson 1994). For the purposes of this study, we defined know-how as organizational strategies and capabilities for implementing and utilizing ERP, know-why as organizational rationales for adopting ERP, and know-what as interpretations of the properties and principles of ERP.

Starting with these main categories, two coders (the coauthors) read the full text of the 100 selected ERP articles in a random order, and developed subcategories for each main category. The subcategories emerged through an iterative process in which each successive article was coded independently, new codes were created as needed, and the two coders then worked to reconcile their evolving coding schemes (Barley et al. 1988; Saunders et al. 2003). A hierarchical coding scheme stabilized after coding
the first 32 articles. These articles were then recoded using the final coding scheme. The coders then coded the remaining 68 articles in three batches. After each batch, inter-coder reliability statistics were calculated for every article and for the batch. Two articles were dropped due to low reliabilities (less than 70 percent), and three others were excluded for not focusing on organizational innovation with ERP. The reliability for the remaining 95 articles ranged from 75 percent to 100 percent, with a mean of 86 percent. Ruling out coding agreements due to chance, the more stringent reliability measure—Cohen’s Kappa (Cohen 1960)—is 0.71 for the 95 articles, which should be interpreted as “good agreement” (Altman 1991, p. 404).

During the coding process, one coder identified the cited contributors for each coded instance of knowledge, using the categories adopter, consultant, journalist, research analyst, and vendor. Once all the articles were coded, the frequencies of every knowledge subcategory, main category, and contributor category in each article were averaged across the two coders (Barley et al. 1988). These frequencies were then divided by the total number of knowledge instances, producing a percentage score for each category for each article.

Community Learning in ERP

Based on our reading of the substance of the discourse, we interpret the discourse curve (Figure 1) as demarking four periods, each with its own distinctive learning pattern: pre-1992 (no articles in the sample set), 1992–1995 (slow growth), 1996–1999 (rapid growth), and 2000–2003 (decline). Although lacking representative articles in the formal data set, the all-important first period could be identified through contributions made by the research analysts who originated the ERP concept.

Period I: Conceptualization (1990–1991)

In the late 1980s, some large manufacturing companies started to express “a great deal of dismay and lack of satisfaction” with the poor functionality in the manufacturing resource planning (MRP II) software offered by IBM, ASK, SSA, and HP (Keller 1999, p. 44). Searching for and evaluating new and unique enterprise software technologies, three research analysts at Gartner created a framework for comparing software packages. They named the framework enterprise resource planning. In April 1990, Gartner first reported its description of the technological environment (see the quotation below) and functionalities of ERP.

[W]e have crafted a checklist for Enterprise Resources Planning (ERP), which we consider the software architecture for the next generation of MRP II….The environmental issues focus on the continuing evolution of computing systems. One area of initial importance will be the implementation of graphical user interfaces in ERP…data will be structured relationally and will be accessed via SQL calls. (Lee Wylie, Gartner, Research Analyst, Computer Integrated Manufacturing, Know-what, Technology Components)

During this period, the discourse on ERP was limited to the analysts’ work on articulating the ERP vision in their reports and at conferences.


After the Gartner analysts introduced the ERP concept, other players quickly joined the discussion, making contributions about what ERP is and how it works.

---

10The codebook is available from the first author.

11The list of these articles is available from the first author.

12Instances from other types of contributors (e.g., academics and industry associations) are negligible.

13ERP first appeared in ABI in October 1991, when a ComputerWorld article mentioned it in passing.

14Features of each quotation are presented in this order: the contributor’s name, role in the community, primary organizational affiliation (if different from publication), publication, main knowledge category, and subcategory.
The goal is a company where information gets entered into a computer once, and only once. A sales rep, say, books an order for gimcracks and enters it into SAP. After that the software makes sure everyone stays informed. When the factory begins assembling the order, shipping can check its progress online. (Ronald Lieber, Journalist, *Fortune*, Know-what, Working Principle)

MRP II vendors “took the label of ERP, slapped it onto their products, and it was off to the races” (Keller 1999, p. 44). Other vendors pondered the design options in producing ERP.

To actually produce an ERP software package, a vendor has a couple of options. One option is to supply customers with a basic information systems infrastructure and then let other companies write programs that work within the system and its standards. Another approach is to bundle together programs that support different business functions into separate “modules.” (Donald A. Hicks, Vendor, Decision Science Inc., *IIE Solutions*, Know-what, Design Principle)

Some vendors, inspired by the new concept, began to offer ERP software packages, including Xerox’s Chess, QAD’s MFG/PRO, and SAP’s R/3.

SAP was founded in 1972 by four former IBM employees in Germany. In 1992, SAP came out with a new version of its software that runs on client-server systems [R/3] instead of mainframes, its revenues climbed from $532 million to an estimated $1.5 billion that year. SAP soon became the market leader of ERP. (Ronald Lieber, Journalist, *Fortune*)

In addition to interpretations of ERP (know-what), rationales for adopting ERP (know-why) began to appear and dominate the ERP discourse in this period. Some early adopters claimed that they adopted ERP to replace their frustrating legacy computing hardware and software. Other early adopters turned to ERP as they faced cost and efficiency pressures.

Easton [a baseball equipment vendor] sells its products on technology, but people are buying looks and price. So we have to make that $60 bat for $20. How do you do it? You reduce set-up times; you reduce waste and scrap; you improve throughput time—whatever it takes. (Kenneth Waltrip, Adopter, Easton, *Manufacturing Systems*, Know-why, to Cut Cost, to Improve Efficiency)

Vendors and journalists sometimes offered know-why on behalf of prospective adopters.

Companies are finding that they require a method of coordinating all the data found in a world wide organization. MRP II cannot cope. (Dinah Greek, Journalist, *Professional Engineering*, Know-why, to Integrate Business)

The discourse initially focused on the know-what and know-why; know-how emerged gradually. Research analysts and vendors led the discussion of know-how by raising the issue of package selection.

We have seen other clients delay ERP implementation because of the lack of robust packages—that has been particularly true of companies with continuous-process or engineer-to-order manufacturing styles. (Erik Keller, Research Analyst, Gartner, *Manufacturing Systems*, Know-how, Fit and Need Determination)

Central to package selection was the trade-off between integrated and “best-of-breed” solutions.

Integrated packages from single vendors are going to require a major up-front investment, but buying different modules from multiple vendors will probably require a lot of effort (read $$$) to get them to work together, and maintenance costs will be higher in the long term. (Shaz Horner, Journalist, Computer Weekly, *IIE Solutions*, Know-how, Best of Breed, End-to-End Solution)

**Period III: Capacitation (1996–1999)**

Major ERP vendors (SAP, Oracle, PeopleSoft, Baan, and J. D. Edwards) enjoyed dramatic growth in the 1990s. By the end of 1998, more than 60 percent of Fortune 1000 companies had implemented ERP core applications (Stein 1999). The ERP fashion
was reaching its full ascendance. Michael Hammer, the eminent reengineering guru, called ERP “the most potent and subversive contemporary instrument of business revolution.” Accompanying the numerous announcements of ERP adoptions and implementations were success and failure stories. Interestingly, horror stories concentrated on just a few firms having implementation disasters.

Texas-based pharmaceutical distributor FoxMeyer Drug actually collapsed following an SAP R/3 implementation, its bankruptcy trustees filed a $500 million lawsuit in 1998 against the German ERP giant [SAP], and another $500 million suit against co-implanter Andersen Consulting. (Malcolm Wheatley, Journalist, CIO Magazine)

Who to blame? Vendors and consultants were criticized for overselling their ERP.

[T]he most revealing moment [at an SAP conference] may have occurred on a bus full of tipsy sales reps. After brief introductions one rep said to the other, “Hey, I heard you finally got [company X] to sign on the dotted line.” Second rep: “Yeah. It took a while for them to see the light, though.” First rep (laughing): “Now comes the darkness.” (Loud laughter throughout bus.) (Michael H. Martin, Journalist, Fortune)

Adopters were criticized for adopting ERP without adequately knowing why and how.

One client I was having lunch with was excited about the ERP implementation soon starting at his company. When I asked him why he was embarking on an ERP program, he looked at me in a puzzled way and said, “No one ever asked me that before.” After 45 minutes of further discussion, he could still not come up with a reason. (Erik Keller, Research Analyst, Gartner, Manufacturing Systems)

However, an abundance of reasons for adopting ERP soon appeared in the discourse. Some adopters addressed on-going struggles such as management control.

But the real reason that many companies are implementing SAP is because management is trying to bring discipline into the organization. (Peter W.C. Mather, Adopter, Air Products and Chemicals Inc., CIO Magazine, Know-why, to Centralize)

Others cited the rise of the Web and the need for ERP as a foundation for e-business. For still others, ERP was a Y2K cure. In this period, vendors’ and journalists’ contributions of know-why declined significantly, leaving adopters the dominant contributors of know-why discussion.

Adopters also dominated in the emergent know-how category. As more and more adopters were implementing and using ERP, they reported their hard-won lessons (from learning-by-doing) to the community, making know-how the largest knowledge category. It included not only package selection tips, but also a host of subcategories on software configuration, organizational change management, project management, and maintenance. For example, when implementing ERP, some companies customized the software to fit their ways of doing business.

The PeopleSoft ERP system didn’t have a data field for the delivery stop sequence, but it needed to relay that information to the warehousing system. To make all the pieces work together, Domino’s decided to take the drastic step of modifying the PeopleSoft software to include these fields. (Jim Krasner, Adopter, Domino’s Pizza Inc., CIO Magazine, Know-how, Customization)

Others preferred refraining from customizing the software by changing their business processes to fit how the software works. Still others suggested a contingency approach.

MBNA used three criteria for accepting or rejecting customization requests: whether the feature existed in its legacy systems and needed to be duplicated, the complexity of the request, and the ROI. (Dick Ho, Adopter, MBNA Corp., Information Week, Know-how, Customization)

“To customize or not” was just one of the know-how topics debated in the ERP community. Discussion of the tradeoff between integrated and best-of-breed solutions (noted earlier) continued. Other frequently contested topics included the extent to which consultants should be used, and how hard project managers should be on deadlines. On the other hand, members of the
community seemed largely to reach consensus on certain other issues; for example, keeping key stakeholders informed was widely accepted as important to implementation success.

As the ERP market grew dramatically, shortage of skilled implementers opened lucrative implementation service opportunities for the IT consulting industry. Consultancies formed partnerships with ERP vendors. With their experiences in helping clients implement ERP, consultants contributed important know-how in the discourse. For example,

> Managing risk within the changes inherent in ERP implementations involves the development of a method to organize, prioritize, and communicate issues before, during, and after changes occur. The management of these changes is broken down into three risk areas. (David Cahn, KPMG, *Manufacturing Systems*, Know-how, Risk Management)

In the meantime, journalists continued scouting among adopters for new ways to innovate with ERP. For example, they found more and more firms “bolting” custom modules on their ERP. Depending on where they drew the boundary of ERP, some called the trend “the second wave of ERP”; others termed it the “post-ERP movement.”

> Now being built onto ERP platforms are applications more attuned to engaging customers and driving profits than the manufacturing, financial, and HR apps that first defined the market. Some of these post-ERP apps—such as sales-force automation and supply-chain management systems—have already emerged. Newer ERP add-ons focus on areas such as demand planning, product data management, and transportation management. (Bruce Caldwell, Journalist, *Information Week*, Know-how, Technology Extension)

The notable increase of know-how primarily contributed by adopters in this period’s discourse indicated that the ERP concept had not only justified, but also capacitated, that is, enabled for developing products, providing services, and use. In contrast, the concept of ERP became increasingly familiar and obvious and, thus, the volume of know-what in the discourse dropped significantly.


In the period since 2000, the ERP product and service market declined, and so too did the discourse (Figure 1). Some vendors ran into serious financial crisis (e.g., Baan). A new round of vendor consolidation began (e.g., PeopleSoft’s acquisition of J. D. Edward and Oracle’s hostile takeover attempt for PeopleSoft in 2003). Some research analysts and journalists believed that the demise of ERP was close. Others continued to capitalize on the ERP concept. In 2001, Gartner introduced ERP II.

Meanwhile, community knowledge on how to best implement and utilize ERP system seemed to have converged on a few rule-like consensuses. For example, go “best-of-breed.” Also, changing business processes was deemed preferable to changing the software code.

> We made it very clear that this would be a business process reorganization and that you couldn’t do it without changing the way you did business. (Jeri Dunn, Adopter, Nestlé USA, *CIO Magazine*, Know-how, Process Change)

Also entering as stock knowledge was the dictum that implementing ERP is not just about installing software: Involve everyone, including the users, business managers, and executives, whose sponsorship and leadership are essential.

By 2003, ERP was no longer at the forefront of IT progress. Instead, it had become the “back-office solution” and the “backbone” for business. The dominant theme in the dwindling discourse maintained by those remaining active in the community—mainly vendors and adopters invested in ERP—was the importance of “recognizing that this is a project that never ends.”

**Evolution of Knowledge**

How did knowledge embedded in the ERP community discourse evolve? Table 1 shows the average percentages of the three main knowledge categories and their contributors by period. The table also shows the significance of change in the percentages between periods. Looking across the periods, know-how increased remarkably, driven primarily by contributions from the
adopters, consultants, and journalists. Although the analysts’ contribution of know-how was dominant during the justification period, it dropped during capacitation. In the justification period, most contributions of know-how were about package selection. Know-how in the later two periods included strategies to manage the implementation and maintenance of ERP systems.

Know-why was the largest category in the justification period. It then decreased but not so substantially as know-what. In fact, adopters’ contributions of know-why increased slightly over the years. Know-what was second in the justification period and became the smallest category later on. Adopters’ contributions of know-what were relatively stable over time, while vendors and consultants reduced their contributions significantly. As to the subcategories of know-what, most subcategories (e.g., design principle and technology component) decreased proportionally with the main category, except the subcategory limitation, which increased slightly.

Table 1. Evolutions of Knowledge Contributed by Community Members*

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Know-How</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>Adopter</td>
<td>22%</td>
<td>60%**</td>
<td>71%*</td>
<td>62%</td>
</tr>
<tr>
<td>Consultant</td>
<td>1%*</td>
<td>30%**</td>
<td>37%*</td>
<td>31%</td>
</tr>
<tr>
<td>Journalist</td>
<td>2%</td>
<td>14%**</td>
<td>15%</td>
<td>13%</td>
</tr>
<tr>
<td>Research Analyst</td>
<td>2%</td>
<td>8%*</td>
<td>11%</td>
<td>9%</td>
</tr>
<tr>
<td>Vendor</td>
<td>5%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
</tr>
<tr>
<td>Know-What</td>
<td>30%</td>
<td>8%**</td>
<td>5%</td>
<td>9%</td>
</tr>
<tr>
<td>Adopter</td>
<td>4%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
</tr>
<tr>
<td>Consultant</td>
<td>7%</td>
<td>1%*</td>
<td>0%</td>
<td>1%</td>
</tr>
<tr>
<td>Journalist</td>
<td>3%</td>
<td>3%</td>
<td>1%</td>
<td>2%</td>
</tr>
<tr>
<td>Research Analyst</td>
<td>3%</td>
<td>0%</td>
<td>0%</td>
<td>1%</td>
</tr>
<tr>
<td>Vendor</td>
<td>13%</td>
<td>1%*</td>
<td>1%</td>
<td>1%</td>
</tr>
<tr>
<td>Know-Why</td>
<td>45%</td>
<td>28%**</td>
<td>23%</td>
<td>27%</td>
</tr>
<tr>
<td>Adopter</td>
<td>14%</td>
<td>15%</td>
<td>18%</td>
<td>16%</td>
</tr>
<tr>
<td>Consultant</td>
<td>7%</td>
<td>5%</td>
<td>1%</td>
<td>3%</td>
</tr>
<tr>
<td>Journalist</td>
<td>10%</td>
<td>5%*</td>
<td>3%</td>
<td>4%</td>
</tr>
<tr>
<td>Research Analyst</td>
<td>0%</td>
<td>2%</td>
<td>1%</td>
<td>1%</td>
</tr>
<tr>
<td>Vendor</td>
<td>13%</td>
<td>1%*</td>
<td>0%</td>
<td>1%</td>
</tr>
</tbody>
</table>

*Percentages may not add up to 100% because some coded knowledge instances belong to other negligible categories (than the three main categories) such as know-when.

Evolution of the Community

Reorganizing the same information, Table 2 shows community members’ contributions of knowledge over time. Vendors’ dominance in the justification period faded in the capacitation and decline periods, as adopters increased their contributions of know-how. The patterns for consultants and journalists were similar. Their overall contributions of knowledge did not change much across the three periods; their increasing know-how contributions were offset by their decreasing contributions of know-what and know-why. Research analysts created ERP in the conceptualization period. Between 1992 and 1995 as others were justifying the ERP concept, analysts focused on know-how.
Table 2. Evolution of Community Members’ Contributions of Knowledge

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td><strong>Adopter</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How</td>
<td>19%</td>
<td>50%**</td>
<td>58%*</td>
<td>51%</td>
</tr>
<tr>
<td>What</td>
<td>4%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
</tr>
<tr>
<td>Why</td>
<td>14%</td>
<td>15%</td>
<td>18%</td>
<td>16%</td>
</tr>
<tr>
<td><strong>Consultant</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How</td>
<td>2%</td>
<td>14%**</td>
<td>15%</td>
<td>13%</td>
</tr>
<tr>
<td>What</td>
<td>7%</td>
<td>1%*</td>
<td>0%</td>
<td>1%</td>
</tr>
<tr>
<td>Why</td>
<td>7%</td>
<td>5%</td>
<td>1%</td>
<td>3%</td>
</tr>
<tr>
<td><strong>Journalist</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How</td>
<td>15%</td>
<td>17%</td>
<td>15%</td>
<td>16%</td>
</tr>
<tr>
<td>What</td>
<td>2%</td>
<td>8%*</td>
<td>11%</td>
<td>9%</td>
</tr>
<tr>
<td>Why</td>
<td>3%</td>
<td>3%</td>
<td>1%</td>
<td>2%</td>
</tr>
<tr>
<td><strong>Research Analyst</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How</td>
<td>15%</td>
<td>7%**</td>
<td>6%</td>
<td>8%</td>
</tr>
<tr>
<td>What</td>
<td>2%</td>
<td>5%*</td>
<td>5%</td>
<td>5%</td>
</tr>
<tr>
<td>Why</td>
<td>3%</td>
<td>0%</td>
<td>0%</td>
<td>1%</td>
</tr>
<tr>
<td><strong>Vendor</strong></td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>How</td>
<td>32%</td>
<td>4%</td>
<td>3%</td>
<td>6%</td>
</tr>
<tr>
<td>What</td>
<td>5%</td>
<td>3%</td>
<td>3%</td>
<td>3%</td>
</tr>
<tr>
<td>Why</td>
<td>13%</td>
<td>1%*</td>
<td>1%</td>
<td>1%</td>
</tr>
</tbody>
</table>

*Percentages may not add up to 100% because some coded knowledge instances were contributed by other types of members in the community, such as academics and industry associations.

bDuring 1992–1995, e.g., adopters contributed on average 19% of the knowledge instances in an article.

cThe 19% adopters’ contribution included 1% know-how, for example.

dSignificance of change from previous period. *p < .05; **p < .01.

To summarize, members of the ERP community played different roles, as indicated by their different knowledge contribution patterns, in the community learning associated with ERP. Most conspicuously, as ERP gained increasing popularity and experienced dramatic market growth, adopters became the leading contributors of knowledge, particularly know-how.

**Situating Community Learning: Toward a Multilevel Process Theory**

In the patterns revealed by the data in Tables 1 and 2, we glimpse the progression in community learning associated with ERP, since “ERP” first entered the lexicon in 1990. Of course, the 95 articles we analyzed represent only a small portion of the total ERP discourse and, as such, afford only partial evidence of the community learning process. Nonetheless, the patterns in question appear in sufficient relief to give us reasonable confidence in theorizing the four-period community learning process for ERP outlined above, including generalizations about the shifting character of the predominant knowledge contributions and the dynamic constitution of the contributing participants.

We summarize our findings in this concluding section against the larger context illustrated by Figure 2. Earlier in this paper we argued that the community learns about a particular IT innovation as its constituent members, individual organizations, learn, because these members actively contribute to the community’s on-going discourse. Reciprocally, these organizations draw on the discourse to learn about the innovation. The result is a multilevel process that, over time, builds up the knowledge available. Figure 2 encapsulates our current view of this overall cycle. The solid lines in the figure correspond to the knowledge flows to
which the current study speaks. The dashed lines represent what we believe are plausible conjectures about other flows making up the cycle. These help set context for interpreting the knowledge flows we have explored here, and they also point to future directions for complementary research.

Conceptualization. In the late 1980s, the discourse on MRP II alerted research analysts at Gartner about MRP II adopters’ increasing dissatisfaction with MRP II (arrow 1 in Figure 2). The analysts then created ERP as the next generation of MRP II, writing reports to explain the concept (arrow 2). Analysts from other research firms and journalists learned about ERP (arrow 1) and offered their interpretations (arrow 2). In this way, ERP was conceptualized.

Justification. More organizations learned about ERP from conferences or from the trade press (arrow 1) and then joined the discussion (arrow 2) about what ERP is (know-what) and why firms should do it (know-why). Some technology vendors began developing and producing ERP products to help instantiate the concept, as they understood it (arrow 3). Based on their learning-by-doing, vendors reported their understanding back to the community (arrow 4). Similarly, certain adopters, after learning about ERP, adopted it and then reported what they learned to the community (arrow 4). The focus of community learning was justification when know-why was the largest category.

Capacitation. By 1996, ERP had been justified, its fashion had been launched, and its market was about to explode. More and more adopters contributed their lessons learned (mostly know-how) to the community discourse (arrow 4). Consultants did the same on a smaller scale. Journalists increased their contribution of know-how (arrow 2), responding to (while helping to promote) an increased demand from their readers. Know-how capacitated ERP, which could then more reliably be produced, serviced, purchased, and used. By the end of 1999, know-what was obvious, know-why became diversified, and know-how dominated the discourse.

Decline. As ERP began to decline, the composition of knowledge in the community discourse did not change much from the previous period, but the volume of knowledge contributions subsided, and the market size for ERP shrank. Knowledge in the community discourse became more coherent, though, as consensus formed on various issues, relating especially to know-how.

The two solid paths in our process model (arrows 2 and 4 in Figure 2) suggest that community learning relies on organizational learning (both learning-about and learning-by-doing). Initially, know-what and know-why coming from analysts’ and journalists’ learning-about help conceptualize and justify the new IT. Later on, know-how from adopters’ learning-by-doing capacitates the IT.
Future research should investigate more activities related to the joint community–organizational learning process, as represented by the dashed lines in Figure 2. Although our data implies that community knowledge was utilized by organizations in their learning-about (arrow 1), we don’t know the details of that process. Further, our data does not say whether community learning influences organizational learning-by-doing (arrow 5) or how learning-by-doing and learning-about interact within organizations (arrows 3 and 6). When organizations learn about new IT, whether and how do they translate knowledge embedded in the community discourse into their own knowledge (Ramiller 2001)? When they learn by doing, whether and how do they translate knowledge encoded in the community discourse into specific processes and practices appropriate for their own organizational contexts (Tillquist 2000)?

Finally, future research should speak more incisively to the quality of community learning, and the processes through which knowledge becomes, or fails to become, validated. Neither community learning nor organizational learning is always successful in generating bonafide knowledge. When an IT is in fashion, if firms jump on the bandwagon without learning much about the IT, the community may fail to incorporate significant, useful knowledge into its discourse. As a consequence, an innovation discourse may be replete with wishful thinking, rather than knowledge.

Our analysis of the ERP discourse demonstrates that knowledge and its contributors can be recorded, classified, and tracked over time. In itself, this affords a new means of support for self-reflectiveness in the innovation practices of the members of a learning community. However, clearly a larger and more ambitious research objective would be a comprehensive learning theory that adds critical assessment to the preliminary synthesis of community learning, learning-by-doing, and learning-by-discourse that we have undertaken here.

Acknowledgments

This study was supported in part by the Information Systems Research Program at UCLA Anderson School of Management. We thank Rob Fichman, Barbara Lawrence, Alex Lopes, Paul Pavlou, Olav Sorenson, Burt Swanson, Lynne Zucker, and the reviewers of ICIS for their helpful comments. We also benefitted from participants of the seminars at UCLA and University of California, Riverside, and of the ISOneWorld 2004 Doctoral Symposium.

References

Slater, D. “An ERP Package for You ... and You ... and You ... and Even You,” *CIO Magazine* (12:9), 1995, pp. 31-36.