

"Sticky Information" and the Locus of Problem Solving: Implications for Innovation

Eric von Hippel

*Sloan School of Management, Massachusetts Institute of Technology, 50 Memorial Drive,
Cambridge, Massachusetts 02139*

To solve a problem, needed information and problem-solving capabilities must be brought together. Often the information used in technical problem solving is costly to acquire, transfer, and use in a new location—is, in our terms, "sticky." In this paper we explore the impact of information stickiness on the locus of innovation-related problem solving. We find, first, that when sticky information needed by problem solvers is held at one site only, problem solving will be carried out at that locus, other things being equal. Second, when more than one locus of sticky information is called upon by problem solvers, the locus of problem solving may iterate among these sites as problem solving proceeds. When the costs of such iteration are high, then, third, problems that draw upon multiple sites of sticky information will sometimes be "task partitioned" into subproblems that each draw on only one such locus, and / or, fourth, investments will be made to reduce the stickiness of information at some locations.

Information stickiness appears to affect a number of issues of importance to researchers and practitioners. Among these are patterns in the diffusion of information, the specialization of firms, the locus of innovation, and the nature of problems selected by problem solvers.

(Sticky Information; Information Transfer Costs; Innovation; User Innovation; Technological Change; Problem Solving; Location of Problem Solving; Iteration)

1. Introduction

To solve a problem, needed information and problem-solving capabilities must be brought together—physically or "virtually"—at a single locus. The need to transfer information from its point of origin to a specified problem-solving site will not affect the locus of problem-solving activity when that information can be shifted at no or little cost. However, when information is costly to acquire, transfer, and use—is, in our terms, "sticky"—we find that patterns in the distribution of problem solving can be affected in several significant ways. In this paper we explore this general matter within the specific context of technical, innovation-related problem solving.

It has not always been clear that technical information used by innovators in the course of their problem-solving work might be costly to transfer from place to place. Indeed, the central tendency in economic theorizing has been to view information as costlessly trans-

ferable, and much of the research on the special character of markets for information has been based precisely on this characteristic. Thus, Arrow observes that "the cost of transmitting a given body of information is frequently very low. . . . In the absence of special legal protection, the owner cannot, however, simply sell information on the open market. Any one purchaser can destroy the monopoly, since he can reproduce the information at little or no cost" (1962, p. 614-615). However, a number of scholars with an empirical as well as theoretical interest in the economics and diffusion of technological information have long argued, and to some extent shown, that the costs of information transfer in technical projects can vary significantly (Nelson 1959, 1982; Rosenberg 1982; Griliches 1957; Mansfield 1968; Pavitt 1987; and Teece 1977).

In this paper we first review and draw on the work of these scholars to provide a reasoned basis for our assumption that information used by technical problem

solvers is in fact often "sticky" (§2). We then explore four patterns in the locus of innovation-related problem solving that appear related to information stickiness. First, when information needed for innovation-related problem solving is held at one locus as sticky information, the locus of problem-solving activity will tend to take place at that site (§3). Second, when more than one locus of sticky information is called upon by problem solvers, the locus of problem-solving activity may move iteratively among such sites as innovation development work proceeds (§4). Third, when the costs of such iteration are high, problem-solving activities that draw upon multiple sites of sticky information will sometimes be "task partitioned" into subproblems that each draw on only one such locus (§5). Fourth, when the costs of iteration are high, efforts will sometimes be directed toward investing in "unsticking" or reducing the stickiness of information held at some sites (§6).

Finally, we will conclude the paper with a discussion of the likely impact of information stickiness on a number of issues of interest to innovation researchers and practitioners. For example, we will reason that the incentives to invest in reducing the stickiness of given information are affected by how frequently that information is a candidate for transfer. Such a pattern would, in turn, offer an economic explanation for a general shift of innovation-related problem solving toward users, as in the current trend in which the producers of software and other products seek to "empower" users by offering them tools that reduce the cost of problem solving and innovation carried out at user sites (§7).

2. "Sticky" Information

As an aid to exploring patterns in the locus of innovation-related problem solving as a function of information transfer costs, we coin the term "sticky" information. We define the stickiness of a given unit of information in a given instance as the incremental expenditure required to transfer that unit of information to a specified locus in a form usable by a given information seeker. When this cost is low, information stickiness is low; when it is high, stickiness is high. Note that in our definition, information stickiness involves not only attributes of the information itself, but attributes of and choices made by information seekers and

information providers. For example, if a particular information seeker is inefficient or less able in acquiring information unit x (e.g., because of a lack of certain tools or complementary information), or if a particular information provider decides to charge for access to unit x , the stickiness of unit x will be higher than it might be under other conditions. The purpose of being inclusive with respect to causes of information stickiness in this definition is to allow us to focus on the impact of information stickiness independent of cause.

As noted earlier, a number of reasons have been advanced and explored as to why information might be sticky. Some reasons have to do with the nature of the information itself, some with the amount of information that must be transferred, and some with attributes of the seekers and providers of the information.

With respect to the impact of the nature of the information to be transferred on variations in information stickiness, consider that some information is encoded in explicit terms, while some is "tacit." Polanyi has pointed out that many human skills, and much human expertise, both extensively employed in technical problem solving, are of the latter sort. He observes that "the aim of a skilful performance is achieved by the observance of a set of rules which are not known as such to the person following them" (Polanyi 1958, p. 49, italicized in original). For example, swimmers are probably not aware of the rules they employ to keep afloat (e.g., in exhaling, they do not completely empty their lungs), nor are medical experts generally aware of the rules they follow in order to reach a diagnosis of various symptoms. "Indeed," Polanyi says, "even in modern industries the indefinable knowledge is still an essential part of technology." And, he reasons, "an art which cannot be specified in detail cannot be transmitted by prescription, since no prescription for it exists. It can be passed on only by example from master to apprentice . . ."—a relatively costly mode of transfer (Polanyi 1958, pp. 52–53).

Rosenberg (1982) argues that drawing on technologically useful information involves not just dealing with theoretical knowledge derived from science, but requires breaking open and examining what transpires "inside the black box" of technological phenomena. Indeed, much technological knowledge is costly, difficult, and slow to diffuse since it deals with "the specific and the

particular," consists of "innumerable small increments . . . ," and may well be tacit (Rosenberg 1976, p. 78). Nelson argues that technological knowledge is "partly a private good and partly a public one," that is, (1) "a set of specific designs and practices," and (2) "a body of generic knowledge that surrounds these and provides understanding of how things work. . . ." (Nelson 1990; pp. 1, 8, 13). The former is often relatively costly and difficult to acquire, learn to use, and diffuse (Nelson 1982), and thus can be private to its creators in certain respects (Nelson and Winter 1982, Chapter 4). In contrast, "generic knowledge not only tends to be germane to a wide variety of uses and users. Such knowledge is the stock in trade of professionals in a field, and there tends to grow up a systematic way of describing and communicating such knowledge, so that when new generic knowledge is created anywhere, it is relatively costless to communicate to other professionals" (Nelson 1990, pp. 11-12).

The cost of transferring information sufficient to solve a given innovation-related problem can also vary according to the amount of information called for by a technical problem solver. Sometimes stickiness is high because a great deal of information with a nonzero transfer cost per unit is drawn upon to complete innovation development work. Thus, successful anticipation and avoidance of all field problems that might affect a new airplane (Rosenberg 1982, Chapter 6) or a new process machine (von Hippel and Tyre 1994) or a new type of laser (Collins 1982) would require that a very large amount of information about the use environment be transferred to the development lab—because one does not know in advance which subset of that information will be relevant to anticipating potential failures. Scientists trying to build a successful copy of a research apparatus often face great difficulties for the same reason. "It's very difficult to make a carbon copy [of a gravity wave detector]. You can make a near one, but if it turns out that what's critical is the way he glued his transducers, and he forgets to tell you that the technician always puts a copy of *Physical Review* on top of them for weight, well, it could make all the difference" (interviewee in Collins 1975, p. 213).

Information stickiness can also be high because organizations must typically have or acquire related information and skills to be able to use the new knowledge

that may be transferred to them. (For example, artists seeking to generate computer art using the mathematics of fractals will not typically be aided by receipt of a software program designed for mathematicians. They must either get the information they seek in "user friendly" form [which in practice means that the transmitter must understand what the recipients already know or can easily learn and must adapt access to the new information accordingly] and/or the recipients must learn the additional complementary information needed to use the existing math program.) Thus, Pavitt points out that "even borrowers of technology must have their own skills, and make their own expenditures on development and production engineering; they cannot treat technology developed elsewhere as a free, or even very cheap, good" (Pavitt 1987, p. 186). Similarly, Cohen and Levinthal argue that a firm's learning or absorptive capacity with respect to new, outside technical information is "largely a function of the firm's prior related knowledge." This stock of knowledge includes not only "basic skills or even a shared language . . ." but also knowledge generated in the course of a firm's own R&D, marketing and manufacturing operations, and technical training programs (Cohen and Levinthal 1990, pp. 128-129). And, again similarly, Evenson and Kislev find in studies of the economic impact of scientific research on agricultural productivity that "little knowledge is borrowed if no indigenous research takes place" (Evenson and Kislev 1973, p. 1314).

Information stickiness can also vary due to other attributes of an information transmitter and receiver. For example, it has been shown that specialized personnel such as "technological gatekeepers" (Allen 1977, Katz and Allen 1982, Katz and Tushman 1980) and specialized organizational structures such as transfer groups (Katz and Allen 1988) can significantly affect the cost of transferring a given unit of information between organizations. And, of course, the decisions of information possessors as to the pricing of access to proprietary information also directly affect the stickiness of that information.

Evidence on the costs of transferring technical information from place to place during innovation-related problem solving also supports the view that technical information can be sticky. A number of empirical studies have been carried out on the costs of transferring a

product or process technology from one firm or location to another with full cooperation on both sides. These show that the costs of information transfer do vary and can be significant. For example, Teece (1977) studied 26 international technology transfer projects and examined the costs of transmitting and absorbing all the relevant unembodied technological knowledge (i.e., information on methods of organization and operations, quality control, manufacturing procedures, and associated information, but not the knowledge embodied in capital goods, blueprints, or technical specifications). He found that transfer costs varied widely for the projects in his sample, ranging from 2 percent to 59 percent of total project costs, and averaging 19 percent—a considerable fraction (Teece 1977; pp. 245, 247).

In sum then, it does appear likely that information sought by technical problem solvers will often be sticky. Therefore it will be useful to examine the effects that information stickiness might have on the locus of innovation-related problem solving. In the following sections we identify four such effects.

3. Sticky Information and the Locus of Innovation-related Problem Solving

When information transfer costs are a significant component of the costs of the planned problem-solving work, it is reasonable that there will be a tendency to carry out innovation-related problem-solving activity at the locus of sticky information, other things being equal—just as, in the case of production, it is reasonable that a firm will seek to locate its factory at a location that will minimize transportation costs, other things being equal.

Evidence bearing on this matter can be found in a number of places. Thus, Rosenberg (1982, Chapter 6) describes "learning by using," which involves problem solving carried out in use environments by, typically, product users. For example, after a given jet engine had been in use for a decade, the cost of maintenance declined to only 30 percent of the initial level because users had learned to perform this task better (Rosenberg 1982, p. 131). Rosenberg argues that such learning by using must be carried out at the user locus because that is the site of the information drawn upon by problem

solvers. Similarly, agricultural researchers seeking to develop new plant varieties that will flourish under given local conditions often find it efficient to shift problem solving to sites where such conditions exist. Griliches (1957), for example, observed that the complex, innovative process of developing hybrid corn seed was carried out separately by local agricultural experiment stations and private seed companies in order to incorporate unique location-specific factors (such as soil type, topography, length of growing season, fertilizer requirements, rainfall, and insect and disease resistance) required in a hybrid for that specific locality.

Finally, Mowery and Rosenberg (1989, Chapter 4) proposed that independent research contractors are most likely to supply research services that exploit little or no firm-specific knowledge because such knowledge is, in our terminology, sticky. To test their hypothesis, they examined the content of all projects carried out by three major independent R&D contracting firms (the Mellon Institute, the Battelle Memorial Institute, and Arthur D. Little, Inc.) between 1900 and 1940. They found that the bulk of the projects carried out by the independent R&D contractors were of a nature that required a relatively small amount of firm-specific knowledge, and reasoned that the projects requiring large amounts of such knowledge had been carried out in client firms' internal labs. This finding is what we would expect if the locus of problem solving is affected by the locus of sticky information.

4. Sticky Information and "Iteration"

When the solving of a given problem requires access to sticky information located at two or more sites, we propose that problem-solving activity will sometimes move iteratively among these sites. We base this proposal on the finding that problem solving in general (Baron 1988, pp. 43–47) and technical problem solving in particular (Marples 1961, Allen 1966) has trial and error as a prominent feature. If and as each cycle of a trial-and-error process requires access to sticky information located at more than one site, it seems reasonable that iterative shiftings of problem-solving activity among sticky information sites will occur as problem solving proceeds (von Hippel 1990a).

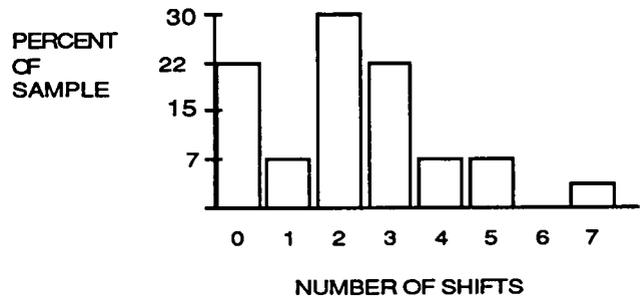
Iteration of the predicted type can often be observed in the problem solving involved in new product and service development. In these arenas two information bases located—at least, initially—in physically different places are typically important for successful problem solving. The first is information on need, located initially with the user. The second is information on solution technologies, located initially at the site of the manufacturer. If need information is sticky at the site of the potential product user, and if solution information is sticky at the site of the product developer, we may see a pattern in which problem-solving activity shuttles back and forth between these two sites.

Thus, as shown schematically in Figure 1, a problem solver may first draw on user need information to generate some attributes for a desired new product or service. Then, manufacturer information may be drawn upon in order to develop a prototype that appears responsive to the specification. The prototype is next tested within its proposed use context to verify function and the accuracy of the initially stated need. If the two do not match satisfactorily—and they often do not—the loci of need and/or capability information must be revisited in search of a closer match. This cycle may be repeated few or many times until an acceptable match is found.

This pattern of iterative shifting of innovation development activity from site to site will be less costly than the transfer of sticky information to a single problem-solving locus given a key condition: The intermediate

Figure 2 Number of Shifts Between Plant and Lab During Problem Solving

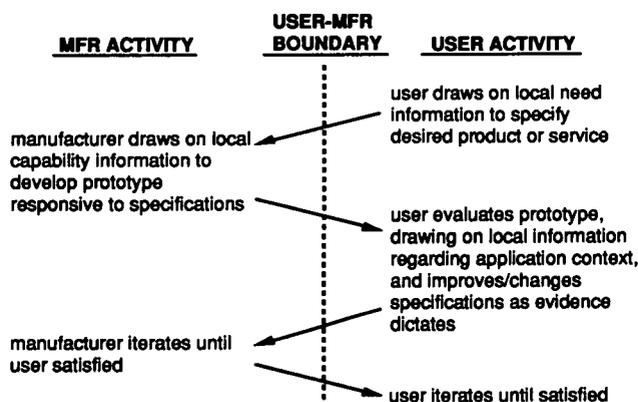
Source: Tyre and von Hippel 1993.



outputs of problem solving conducted at each locus that are transferred between sites must be less sticky than the information operated upon to produce the outputs. Intuitively it seems reasonable that this will often be the case: Such an intermediate output may be in the form of nonembodied information transferable at low cost, or it may be in the form of a prototype that can be economically transferred. For example, an artist may not be able to transfer all information involved in the creative process that brings him or her to specify to a supplier, "I need a green paint of precisely X hue and luminance." However, that (nonembodied) need specification is very simple and precise, and it can be transferred at very low cost. Similarly, the responding paint manufacturer may be able to create and transfer the requested shade of green to the artist (embodied in a prototype or final product), but not be able to transfer the complex knowledge drawn on by that firm's chemists to achieve the feat.

Recent empirical studies report the iterative problem-solving pattern described in Figure 1. Tyre and von Hippel (1993) explored the innovation-related problem solving involved in identifying and diagnosing 27 field failures in process equipment used to automatically assemble complex circuit boards. They observed repeated shifts in the locus of technical problem-solving activity occurring during this work, with the number of shifts found ranging from 0 to 7, and averaging about 2.3 times per problem identified and diagnosed (Figure 2). These shifts involved engineers traveling back and forth between development lab and plant (two to three hours by car), carrying out technical problem-solving activities

Figure 1 Iterative Problem-solving Pattern Often Encountered in New Product and Service Development



at each site, and carrying intermediate findings back and forth in their minds and/or computer data disks. For example, to begin the diagnosis of a machine that was failing in the field, the designers of that particular machine would often visit the plant where it was being used in order to observe the malfunction in context and run diagnostic tests. Then they would return to the development lab (the site of specialized lab equipment, relevant expertise, and other types of information) to examine the test results and continue their diagnostic work. Often this work would lead to the need for a second trip to the field for more data collection, and so forth.

In this study, the cost of the iterative shifting of innovation-related problem-solving activity observed did appear to be less than the cost of transferring all information needed by technical problem solvers to a single locus—say, the development lab. While no particular item of information found useful for diagnosing a particular process machine malfunction was very costly to transfer from the plant to the development lab, the specific items needed by the lab could not be identified without problem-solving and trial-and-error activities conducted in the plant. As a consequence, shifting all information needed to diagnose field problems from plant to lab would have meant shifting a great deal of information—effectively the entire use environment—from plant to lab (von Hippel and Tyre 1994). Carrying out such a massive information transfer would have been much more costly than the iterative transfer of problem-solving activity between plant and lab that was found to have taken place.

In studying product innovation in the Danish food industry Kristensen (1992) observed a similar iterative pattern. Here, information is passed back and forth between Danish food producers and customers located in culturally, linguistically, and geographically distant markets. Often, prototypes are used as the medium for information transfer because, as Kristensen points out, "prototypes are not only inexpensive and fast to produce in the food products industry; they are also small and inexpensive to transport." When, for example, a Danish bakery firm was asked to develop a new frozen unbaked cake by a British retail food chain, the bakery's production department responded by developing several prototypes of the proposed cake and sending them to the customer to bake, taste, and smell, and to evaluate

on the basis of "local tastes and the situation they were meant for—a type of social gathering not practised in Denmark." Comments on the baked cakes were sent back to the producer, who adjusted the recipes accordingly, "using his familiarity with baking and with local raw materials." In total, "five successive revised generations were sent during the course of three months before the Danish producer and the United Kingdom retail chain's test kitchen reached the generation of satisfactory variations." Kristensen reports that over 40 percent of the 103 Danish food producers he studied had developed one or more products within the previous two years via such iterative interactions with customers (Kristensen 1992; pp. 204–205, 210).

The likely ubiquity of the iteration pattern we describe is suggested by the recent emergence of product development procedures specifically designed to implement such a pattern. For example "rapid prototyping" is a method of software development explicitly designed to shuttle repeatedly between manufacturer and users, replacing the traditional, specification-driven ("waterfall") method of software development. In that traditional method, systems analysts meet with users at the start of a project to determine user needs and agree on a written product requirements specification, and they then work isolated from further user contact until the completed product is delivered (in six to eight months or up to two years or more), all too often "late, over-budget and not what the customer wanted" (Zelkowitz 1980, p. 1037). In the rapid prototyping method, manufacturers respond to initial user need inputs by quickly developing and delivering to users (usually within weeks) an inexpensive, easy to modify, working model that simulates a lot of the functionality of the proposed new software. Users then learn by using the prototype in their own setting on their own data and clarify their needs, in part by drawing on their tacit knowledge and experience (Gronbaek 1989, pp. 114–116). Users then relay requests for changes or new features to the software developers, who respond by drawing on their own sticky information and tools to make modifications to the prototype. Some of these modifications are minor, such as altering report formats, and some are major, such as implementing a new feature or modifying the basic structure of the prototype (Feld 1990, p. 14). A revised prototype is then sent to the user, and this process of iteration between developer and user is repeated

until an acceptable fit between need and solution is found. A number of individual case studies and experiments have shown that rapid prototyping methods are not only less costly than traditional, noniterative methods but are able to "better satisfy true user requirements and produce information and functionality that is more complete, more accurate, and more meaningful" (Connell and Shafer 1989, p. 15; Boehm et al. 1984; Gomaa 1983).

5. Sticky Information and "Task Partitioning"

When more than one locus of sticky information must be drawn upon to solve a problem, common experience suggests that even iteration can sometimes be very costly with respect to time and effort. For example, no patient likes the shuffling back and forth and time lags involved when a medical condition involves even routine diagnostic tests by and coordinated problem solving among several physicians in different specialties. And, similarly, no designer likes the cost in time and money and frustration involved in repeated redesign of a finished product or service as a result of new information uncovered in the course of test marketing conducted at user sites.

As a consequence, we reason that when the information transfer costs of iteration are high, innovation-related problem-solving activities that require access to multiple loci of sticky information will sometimes be "task partitioned" into subproblems that each draw on only one such locus of sticky information. Because there are many different ways to partition a given innovation project, the selection of a particular partitioning can have a very strong effect on how much information from one task must be drawn upon to solve another as the technical problem-solving work progresses (von Hippel 1990b). As a schematic illustration, consider two alternative ways of partitioning the project of designing a new airplane:

—"Firm X is responsible for the design of the aircraft body, and firm Y is responsible for the design of the engine,

and:

—"Firm X is responsible for designing the front half of both the aircraft body and engine, and firm Y is responsible for designing the back half of both.

Taken together, each of these proposed task partitionings has the same project outcome—a complete aircraft design. But the two differ greatly with respect to the level of information exchange and/or iterative relocation of problem-solving activities. Clearly, information transfer costs would be much higher in the second alternative than in the first: Many design decisions affecting the shape of the "front half" of an aircraft body would force related changes on the designers of the "back half" of the body, and vice versa, because the two halves cannot be considered independently with respect to aerodynamics.

As a real world example of the task partitioning of an innovation project, consider the problem-solving work involved in designing a silicon integrated circuit on a semiconductor chip for a custom application. In this design problem, two sticky data bases are central to the problem-solving work: (1) information at the circuit user locus involving a rich and complex understanding of both the overall application in which the custom integrated circuit will play a role and the specific function required of that circuit; (2) information at the circuit manufacturer locus involving a rich and complex understanding of the constraints and possibilities of the silicon fabrication process that the manufacturer uses to produce integrated circuits.

Traditionally, custom integrated circuits were developed in an iterative Figure 1-like process between a circuit user possessing sticky need information and an integrated circuit manufacturer possessing sticky information about designing and producing silicon integrated circuits. The process would begin with a user specifying the functions that the custom chip was to perform to a circuit design specialist employed by the integrated circuit manufacturer. The chip would then be designed at the manufacturer locus, and an (expensive) prototype would be produced and sent to the user. Testing by the user would typically reveal faults in the chip and/or the initial specification, responsive changes would be made, a new prototype built, and so forth.

More recently, the Application Specific Integrated Circuit (ASIC) method of making custom integrated circuits has come into wide practice. In the ASIC method, the overall problem of designing custom circuits is partitioned into subproblems that each draw on only one locus of sticky information, thereby eliminating the need to iterate between two such sites in the

design process. The manufacturer of ASICs draws on its own sticky information to develop and improve the fabrication processes in its manufacturing plant, a "silicon foundry." The manufacturer also draws on its own sticky information to design "standard" silicon wafers that contain an array of unconnected circuit elements such as logic gates. These standard circuit elements arrays are designed by the manufacturer to be interconnectable into working integrated circuits by the later addition of custom interconnection layers designed in accordance with the needs of specific users. To facilitate this user task, the manufacturer provides custom circuit users with a user-friendly Computer-Aided Design (CAD) software package that enables them to design a custom interconnection layer that will meet their specific application needs and yet stay within the production capabilities of the manufacturer's silicon foundry. This CAD software also allows the user to simulate the function of the custom circuit under design, and to conduct trial-and-error experiments. Taken together, these capabilities allow the user to both design a circuit, and to refine need specifications and the desired circuit function through an iterative process that draws only on sticky information located at the user site. In sum, by partitioning the overall circuit design task somewhat differently than is done in the traditional method, the ASIC method of designing custom integrated circuits reduces the need for the iterative shifting of the locus of innovation-related problem solving between user and manufacturer.

6. Sticky Information and Investing in "Unsticking" Information

The stickiness of a given body of information is not immutable. Thus, when the costs of iteration are considered to be high, efforts will sometimes be directed toward investing in "unsticking" or reducing the stickiness of some of the information. For example, firms may reduce the stickiness of a critical form of technical expertise by investing in converting some of that expertise from tacit knowledge to the more explicit and easily transferable form of a software "expert system" (Davis 1986). Or they may invest in reducing the stickiness of information of interest to users by converting it into a remotely accessible and user-friendly computer data base. This is what the travel industry did, for ex-

ample, when it invested substantial sums to put its various data bases for airline schedules, hotel reservations, and car rentals "on-line" in a user-accessible form.

However, incentives to unstick information can vary. For example, suppose that to solve a particular problem, two units of equally sticky information are required, one from a user and one from a manufacturer. In that case, there will be an equal incentive operating to unstick either of these units of information in order to reduce the cost of transfer, other things (such as the cost of unsticking) being equal. But now suppose that there is reason to expect that one of the units of information, say the manufacturer's, will be a candidate for transfer n times in the future, while the user's unit of information will be of interest to problem solvers only once. For example, suppose that a manufacturer expects to have the same technical information called on repeatedly to solve n user product application problems and each problem involves unique user information. In that case, the total incentive to unstick the manufacturer's information across the entire series of user problems is n times higher than the incentive for an individual user to unstick its problem-related information. And, as an important consequence, it is reasonable that the locus of problem-solving activity will tend to shift to the locus of the less frequently called-upon information—in the case of our example, to the user.

As illustration, recall the shift from the traditional iterative method of designing custom integrated circuits to the ASIC task-partitioning method that we described earlier. During the problem-solving work of circuit design, each circuit designer requires access to the same information about the constraints of the circuit manufacturing process, but requires different information about the specific application being designed for. As a consequence, the ASIC manufacturer found it economic to unstick the repeatedly called-upon production process information by investing in encoding it in a user-friendly CAD package. And, as a further consequence, the problem-solving activity of custom circuit design was shifted to the locus of sticky information regarding each unique application—the user.

The particular pattern just described will often hold in real-world problem solving, we suggest, because it offers a way for manufacturers to seek economies of scale by producing standard products, while at the same time enabling users to carry out the problem solving

needed to adapt these to specific local needs and conditions. Consider, for example, the current trend in software (Feld 1990) toward "empowering users." To empower users, manufacturers invest in unsticking some of their programming expertise and information by offering user-friendly programming languages such as Object Oriented Programming (OOPs), and user-tailorable application programs or tool boxes. This has the effect of shifting the problem-solving activity involved in tailoring software to local conditions to the locus of sticky information regarding those local conditions—the user.

7. Discussion and Suggestions for Further Research

In this paper we have begun to explore the impact of sticky information on the locus of innovation-related problem solving, and we propose that further study of information stickiness can be of significant value and interest to both innovation researchers and innovation practitioners.

In the course of our initial work we have observed and discussed four patterns in the distribution of innovation-related problem solving associated with efforts made by technical problem solvers to reduce information transfer costs. First, when technical information that is costly to acquire, transfer, and use is held in one locus of sticky information, innovation-related problem-solving activities will tend to move to that locus; second, when more than one locus of sticky information is called upon to solve a problem, the locus of problem-solving activity will tend to iterate among these sticky information sites as innovation development work proceeds; third, when the costs of such iteration are high, innovation-related problem-solving activities that require access to multiple loci of sticky information will sometimes be "task partitioned" into subproblems that each draw on only one such locus of sticky information; and fourth, when the costs of iteration are high, investments may be made toward investing in "unsticking" or reducing the stickiness of information held at some sites.

This short list is not intended to be exhaustive, and further work may identify additional patterns as well as usefully elaborate on the four already identified. For example, in the present paper we have not examined patterns in the distribution of innovation-related prob-

lem solving that will be visible when a problem *can* be solved using only technical information that can be acquired, transferred, and used without cost or nearly so. We speculate that, in such cases, the locus of problem-solving activities will depend on the costs associated with locating the noninformation components necessary to the technical problem-solving work. Problems appropriate for problem solvers who "telecommute" can fall into this category because the data inputs and outputs called upon can be sent nearly anywhere at low cost over telecommunication networks. Therefore, telecommuters can locate themselves wherever they and their employers find it most cost-effective and convenient to carry out their problem-solving work.

Innovation practitioners may wish to use the information transfer patterns we have discussed in this paper to consciously manage their information transfer costs. The value of doing this in any particular circumstance will depend on these strategies not adversely affecting other innovation cost factors or an innovator's abilities to appropriate innovation-related benefit. We think this can often be the case even though, on the face of it, the latter condition seems problematic. After all, patents and trade secrecy and lead time can all be important to an innovator's ability to profit from an innovation (von Hippel 1988, Chapter 5), and these all depend on an innovator's maintaining at least some secrecy at least for a while. But how can one expect an innovating firm to keep secrets if it conducts problem solving not on the innovator's premises but at sites of sticky information? For example, would not a firm that wants to keep a chemical formula a trade secret be ill-advised to conduct some of the technical development work at a customer site?

Often, however, conducting innovation-related problem solving at remote sites need not compromise an innovator's ability to protect commercially important secrets. First, consider that firms can come to some legal arrangement that will maintain secrecy for problem solving done at another locus. Second, consider that firms routinely do locate some of this type of problem solving off their premises without taking legal precautions, and with no apparent impact on their ability to appropriate benefit from their innovations. In some instances, this is explicable because the innovation development task undertaken outside the firm is just a piece of the whole, and revealing a part does not reveal

the whole to would-be imitators. Thus, firms routinely ask outside suppliers to develop components of an innovative product, engage in market research and product testing on customer sites, and so forth. In other instances, an innovation being worked upon without benefit of secrecy is nonetheless protected because it is tied in some way to a product or service or process that is protected. Thus, a supplier of a proprietary computer program may benefit from nonproprietary improvements to it, because the improvements will only operate in conjunction with the proprietary program.

The concept of information stickiness can also enable us to understand more about patterns of specialization among individuals and firms. Since an organizational boundary can add to the costs of information transfer, it seems likely that firms seeking to economize with respect to the transfer of sticky information will seek to align their organizational boundaries—and their specializations—with the partitionings dictated by the types of innovation-related problem-solving tasks that are important to them. For similar reasons, consideration of the impact of sticky information may be useful in studying the various collaborative innovation patterns that are being practiced by firms today (e.g., Gemunden 1980). We also propose that studies of sticky information can increase our understanding of how firms protect, sell, trade, diffuse, and appropriate benefit from information. Thus, stickiness can help the possessors of valuable information to prevent unintentional diffusion to competitors, but that same property may make it more costly to diffuse the information intentionally.

Studies that use information stickiness as a variable can also help researchers to explore patterns in the nature of problems selected by technical problem solvers. It seems reasonable that problems that involve low information transfer costs would tend to be selected preferentially. Thus, a firm may elect to develop new products that draw on local information in preference to those that require costly information transfers from suppliers or users or others. Similarly, responses to information transfer costs, such as a decision to partition problem-solving tasks in a different way, or to unstick certain information, can affect the kind of solutions that technical problem solvers may develop to a given problem. For example, the development of single-site "desktop publishing" (which removed the need for iterative problem solving among author, graphic designer, and

printer) may well enable author/"publishers" to create very different documents as well as less expensive ones. And the development of home medical diagnosis kits (which reduce the need for information transfers among patient, doctor, and medical laboratory) may bring about qualitative and quantitative changes in the type of medical care that is demanded and the way it is delivered.

Finally, it is interesting to speculate about the patterns in the locus of innovation-related problem solving that will emerge as the computerization of problem-solving activities continues to make information even more accessible via computer networks and increasingly portable, inexpensive, and user-friendly computer equipment and software. Taken together, these trends can certainly facilitate "anywhere" problem solving when all of the information drawn upon to solve a technical problem is nonsticky, as in the instance of telecommuting discussed above. When information transfer costs vary and at least some of the needed information is sticky, however, these same trends can make the patterns discussed in this paper even more salient. Thus, researchers equipped with computers and network access will be free to transfer their work to and among field sites containing sticky information, managers will be free to move decision making to the sites of critical tasks, and product designers will be free to design products working directly with users at user sites. It will be an interesting world to develop and explore!¹

¹ I am very grateful to my colleagues Anne Carter, Bradley Feld, Dietmar Harhoff, Zvi Griliches, Ralph Katz, Richard Nelson, Nathan Rosenberg, Stephan Schrader, Stefan Thomke, Marcie Tyre, and Jessie von Hippel for their contributions to the ideas explored in this paper.

I thank the Sloan Foundation for funding the research reported on in this paper.

References

- Allen, Thomas J., "Studies of the Problem-Solving Process in Engineering Design," *IEEE Transactions on Engineering Management*, EM-13, 2, June (1966), 72-83.
- , *Managing the Flow of Technology: Technology Transfer and the Dissemination of Technological Information Within the R&D Organization*, MIT Press, Cambridge, MA, 1977.
- Arrow, Kenneth J., "Economic Welfare and the Allocation of Resources of Invention," in Richard R. Nelson (Ed.) *The Rate and Direction of Inventive Activity: Economic and Social Factors*, A Report of the National Bureau of Economic Research, Princeton University Press, Princeton, NJ, 1962, 609-625.

- Baron, Jonathan, *Thinking and Deciding*, Cambridge University Press, New York, 1988.
- Boehm, Barry W., Terence E. Gray, and Thomas Seewaldt, "Prototyping Versus Specifying: A Multiproject Experiment," *IEEE Transactions on Software Engineering*, SE-10, May (1984), 290-303.
- Cohen, Wesley M. and Daniel A. Levinthal, "Absorptive Capacity: A New Perspective on Learning and Innovation," *Administrative Sci. Quarterly*, 35, 1, March (1990), 128-152.
- Collins, H. M., "Tacit Knowledge and Scientific Networks," in Barry Barnes and David Edge (Eds.), *Science in Context: Readings in the Sociology of Science*, MIT Press, Cambridge, MA, 1974, 1982, 44-64.
- , "The Seven Sexes: A Study in the Sociology of a Phenomenon, or the Replication of Experiments in Physics," *Sociology*, 9, 2, May (1975), 205-224.
- Connell, John L. and Linda Brice Shafer, *Structured Rapid Prototyping: An Evolutionary Approach to Software Development*, Prentice-Hall, Englewood Cliffs, NJ, 1989.
- Davis, Randall, "Knowledge-Based Systems," *Science*, 231, 4741, February 28 (1986), 957-963.
- Evenson, Robert E. and Yoav Kislev, *Agricultural Research and Productivity*, Yale University Press, New Haven, CT, 1975.
- Feld, Bradley A., "The Changing Role of the User in the Development of Application Software," Working Paper No. BPS 3152-90, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, August 1990.
- Gemunden, Hans Georg, "Efficient Interaction Strategies in Marketing of Capital Goods" (in English), Working Paper, Institut für Angewandte Betriebswirtschaftslehre und Unternehmensführung, University of Karlsruhe, Karlsruhe, Germany, n.d. Published as "Effiziente Interaktionsstrategien im Investitionsgütermarketing," *Marketing ZFP*, March (1980), 21-32.
- Gomaa, Hassaa, "The Impact of Rapid Prototyping on Specifying User Requirements," *ACM Sigsoft Software Engineering Notes*, 8, 2, April (1983), 17-28.
- Griliches, Zvi, "Hybrid Corn: An Exploration in the Economics of Technical Change," *Econometrica*, 25, 4, October (1957), 501-522.
- Gronbaek, Kaj, "Rapid Prototyping with Fourth Generation Systems—An Empirical Study," *Office: Technology and People*, 5, 2, September (1989), 105-125.
- Katz, Ralph and Thomas J. Allen, "Investigating the Not Invented Here (NIH) Syndrome: A Look at the Performance, Tenure, and Communication Patterns of 50 R&D Project Groups," *R&D Management*, 12, 1, January (1982), 7-19.
- and —, "Organizational Issues in the Introduction of New Technologies," in Ralph Katz (Ed.), *Managing Professionals in Innovative Organizations*, Ballinger, Cambridge, MA, 1988, 442-456.
- and Michael L. Tushman, "External Communication and Project Performance: An Investigation into the Role of Gatekeepers," *Management Sci.*, 26, 11, November (1980), 1071-1085.
- Kristensen, Preben Sander, "Flying Prototypes: Production Departments' Direct Interaction with External Customers," *International J. Operations and Production Management*, 12, 7, 8 (1992), 195-211.
- Mansfield, Edwin, *Industrial Research and Technological Innovation: An Econometric Analysis*, W. W. Norton, New York, 1968.
- Marples, David L., "The Decisions of Engineering Design," *IRE Transactions on Engineering Management*, June (1961), 55-71.
- Mowery, David C. and Nathan Rosenberg, *Technology and the Pursuit of Economic Growth*, Cambridge University Press, New York, 1989.
- Nelson, Richard R., "The Simple Economics of Basic Scientific Research," *J. Political Economy*, 67, June (1959), 297-306.
- , "The Role of Knowledge in R&D Efficiency," *Quarterly J. Economics*, 97, 3, August (1982), 453-470.
- , "What is Public and What is Private About Technology?" Consortium on Competitiveness and Cooperation, Working Paper No. 90-9, Center for Research in Management, University of California at Berkeley, Berkeley, CA, April 1990.
- and Sidney G. Winter, *An Evolutionary Theory of Economic Change*, Harvard University Press, Cambridge, MA, 1982.
- Pavitt, Keith, "The Objectives of Technology Policy," *Science and Public Policy*, 14, 4, August (1987), 182-188.
- Polanyi, Michael, *Personal Knowledge: Towards a Post-Critical Philosophy*, University of Chicago Press, Chicago, IL, 1958.
- Rosenberg, Nathan, *Perspectives on Technology*, Cambridge University Press, New York, 1976.
- , *Inside the Black Box: Technology and Economics*, Cambridge University Press, New York, 1982.
- Teece, David J., "Technology Transfer by Multinational Firms: The Resource Cost of Transferring Technological Know-How," *Economic J.*, 87, 346, June (1977), 242-261.
- Tyre, Marcie J. and Eric von Hippel, "Locating Adaptive Learning: The Situated Nature of Adaptive Learning in Organizations," Working Paper No. BPS 3568-93, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, May 1993.
- von Hippel, Eric, *The Sources of Innovation*, Oxford University Press, New York, 1988.
- , "The Impact of 'Sticky Information' on Innovation and Problem-Solving," Working Paper No. BPS 3147-90, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, April, 1990a (revised October 1991).
- , "Task Partitioning: An Innovation Process Variable," *Research Policy*, 19, 5, October (1990b), 407-418.
- and Marcie J. Tyre, "How Learning by Doing Is Done: Problem Identification in Novel Process Equipment," *Research Policy*, forthcoming, (1994).
- Zelkowitz, Marvin V., "A Case Study in Rapid Prototyping," *Software—Practice and Experience*, 10, 2, December (1980), 1037-1042.

Accepted by Ralph Katz; received January 1992. This paper has been with the author 11 months for 2 revisions.

Copyright 1994, by INFORMS, all rights reserved. Copyright of Management Science is the property of INFORMS: Institute for Operations Research and its content may not be copied or emailed to multiple sites or posted to a listserv without the copyright holder's express written permission. However, users may print, download, or email articles for individual use.