Journal
Berkeley Planning Journal, 10(1)

Author
Blanco, Hilda

Publication Date
1995

DOI
10.5070/BP310113058

PRAGMATIC KNOWLEDGE CODES

Hilda Blanco

The article explains a research program that stems from the author's recent book, How to think about social problems *(1994), where she argues for a reorganization of the domains of knowledge in public policy and planning into explicit, pragmatic knowledge codes. The author argues that knowledge in the public policy and planning fields is the common knowledge necessary for informed and responsible participation in public affairs, and thus a necessary condition for creating participatory, democratic communities in modern society.*

The research project Thalia, outlined here, aims to show how expert knowledge in a relatively simple urban planning knowledge domain, urban forestry, can be made explicit and simulated. Thalia involves the application of an artificial intelligence cognitive architecture, FORR (FOr the Right Reasons), developed by computer scientist Susan Epstein. FORR is an architecture particularly promising for public policy and planning because of its ability to incorporate pluralism and pragmatism.

The Rationale

The research I am embarking on stems from one of the major theses of my recent book, *How to think about social problems* (1994), where I argue for a reorganization of the domains of knowledge in public policy and planning into explicit, pragmatic knowledge codes.[1] The project Thalia, outlined here, aims to provide proof of the thesis by making explicit and organizing the knowledge in a relatively simple knowledge domain in urban planning, urban forestry. I propose to do this by developing a computer simulation of expert knowledge in urban forestry.

The rationale for this project is as follows: academic knowledge in urban planning, as in most professions, is presented in a "peculiarly disassembled" way that impedes its use (Abbott 1988, 53). This is partially due to the fact that professions rely on apprenticeships and experience to develop expertise, since much of the professional knowledge is tacit procedural knowledge instead of declarative propositional knowledge – i.e., knowing *how* rather than knowing *that*. This type of procedural knowledge is not easily made explicit, especially through written materials. Moreover, maintaining professional knowledge as tacit knowledge serves the interest of professions; to the extent that professional knowledge remains opaque, monopoly over knowledge can be sustained, and professional jurisdiction retained.

This state of affairs may be acceptable for most professions, but, as I argue in my book, it is objectionable for urban planning and the other public policy fields. The knowledge of public policy and planning in general, and urban planning in particular, is the common knowledge necessary for informed and responsible participation in public affairs. Making such tacit knowledge explicit in a practical form is a necessary, although not sufficient, requirement for creating participatory, democratic communities in modern society.[2]

I call for the reorganization of knowledge in public policy and planning into pragmatic knowledge codes (Blanco 1994, 167-80). The term "pragmatic knowledge code" refers to a knowledge domain organized for practical use that responds to the three essential acts of professional practice: diagnosing, inferring, and treating (Abbott 1988, 40). According to Abbott, in his seminal work on the nature of professions, the task of professional inference lies in the middle ground between diagnosis and treatment and relates professional knowledge based on antecedents to the peculiarities of the presenting client or situation. The inference required can vary from very little in routine cases, to extensive in cases where the connection between diagnosis and treatment is obscure (id. 49).

Employing Umberto Eco's (1979, 32-40) theory of codes, I argue that pragmatic knowledge codes are composed of four sets of systems:

a. situations – i.e., states of the world;

b. problem identification – i.e., assessments or meanings of (a);

c. strategies or behavioral responses to (b);

d. a set of more or less loose and extensive rules relating (a) to (b) and (b) to (c).

In policy and planning, the systems in (a) and (b) are correlated as diagnostic indicators. System (d), which relates systems (a), (b), and (c), contains rules of varying looseness and logical length, which may employ simple or elaborate models, empirical generalizations, and theories.

Eco's notion of codes throws light on different types of problems. Technical problem-solving, the kind associated with tame problems in Rittel and Webber's tame-wicked distinction (1973), I believe, can be construed as involving coded inferences. The unproblematic, given nature of these problems is due to social conventions that accept and recognize the expertise of a number of technical professions in our society. For example, perceiving the clogging of highways with bumper-to-bumper cars traveling at 15 miles per hour as a traffic congestion problem, which requires the expertise of transportation engineers, is the result of a convention that sanctions the application of their professional code to certain types of situations.

An Illustration

To illustrate the notion of pragmatic knowledge codes, consider, for example, the subject of water quality.[3] Water quality problems are identified by a set of diagnostic indicators. Although the Federal Water Pollution Control Administration monitors over 400 parameters, five characteristics are typically used to determine water purity: coliform bacteria, dissolved oxygen, nitrate, phosphorus, and suspended sediments (Commoner 1987, 5). For illustration purposes, we will focus on coliform bacteria (Figure 1).

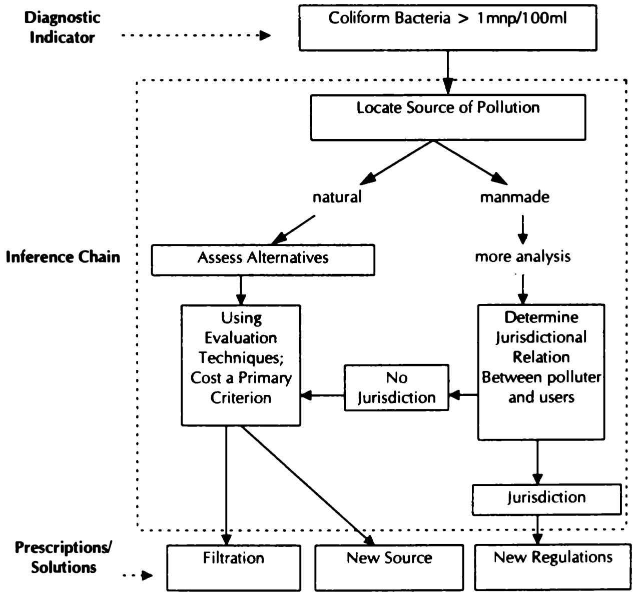
The diagnostic indicator or standard used to determine whether a water quality problem exists (in this case, the presence of coliform bacteria above 1 per 100 ml using the MPN (most probable number) sampling technique) is supported by environmental/medical research that links the effects of coliform bacteria to other organisms, including humans, as well as by a complex political/legal/institutional framework. New ecological research may support a more stringent standard; more laissez-faire political administrations may set a looser standard.

When a problem is perceived, the major types of prescriptions or solution strategies are filtration, finding a new cleaner source, and new regulations. A complex chain of reasoning often leads from the application of a diagnostic indicator that identifies a problem to a particular solution. Part of the inferential chain requires information on the details of the problem. In this case, the chain of reasoning involves first determining the source of pollution, and in particular, whether the source is natural or man-made. If it is natural, then determining whether it would be better to filter or to find a new cleaner source of water involves an assessment process. This assessment of alternatives could take the form of a specific evaluation technique, such as cost-benefit analysis. In any case, cost is a primary if not the primary criterion used.

On the other hand, if the source of pollution is man-made, the relationship between the governmental jurisdictions that represent polluters and water users becomes important. If the governmental body that perceives the existence of a water quality problem has jurisdiction over the polluters, then the most direct solution is to enact regulations to reduce coliform bacteria to acceptable levels. If the governmental body lacks that jurisdiction, then the primary options are to filter or to find a new source – i.e., the same alternatives available for natural sources of pollution. Of course, there are often multiple sources of pollution, in which case the process of ascertaining the superior option would be more complex, and the solution strategy is likely to involve a combination of techniques.

Figure 1

Drinking Water Quality
An Illustration of Elements in Pragmatic Knowledge Codes



Pragmatic knowledge codes organize knowledge according to steps in the rational planning model. Diagnostic indicators typically identify problem severity, as well as normal or non-problematic states. They incorporate both the problem identification and the goals formulation steps of the rational model. The inferential chain from diagnostic indicators to recommended strategy includes an assessment process that takes into account the peculiarities of the problem presented. This step corresponds to the assessment of alternatives in the rational model.

Pragmatic knowledge codes, however, would more fully articulate the inferential chain and tag it with research sources and with political, legal, and institutional sources. For example, in this case, political jurisdiction plays an important role in determining a solution for man-made pollution problems. A gloss on political jurisdictions in this country, explaining the three-tiered system of government, the power of the states, home rule by localities, the rigidity of political boundaries, and the uses of special districts, would be important to understanding the feasibility of the various alternatives. The solution strategies would be set out in detail, including variations, frequency of use, implementation problems, and results of any evaluations.

Organizing knowledge in this way supports a pragmatic view of human nature, experience, and knowledge, since codes are purposeful organizations of perception, thought, and action with a clearly practical intent. However, this is not the way most knowledge in public policy and planning is currently organized. For example, the best known text in land use planning (Kaiser, Godschalk, and Chapin 1995) has no substantive discussion of zoning mechanisms, the chief means of implementing land use planning in this country.

Pragmatic Knowledge Codes and Artificial Intelligence

I arrived at the notion of pragmatic knowledge codes through my work articulating the relation of American pragmatism to public policy and planning. Specifically, the notion is inspired by John Dewey's ideas concerning the need to create intelligent publics to further democratic practice (Dewey 1927). My interest is in making expert knowledge in these fields explicit and accessible to the public, particularly to community organizations and schools, as the basis for civic education (Blanco 1993).

I then realized that the concept of expert systems in computer science is very similar to my notion of pragmatic knowledge codes. Expert systems are meant to capture both the knowledge base and the inferential mechanisms that experts use to make routine decisions (Michie 1982; Edmunds 1988; Gupta and Prasad 1988). Because of their transparency, expert systems are a particularly promising way to organize knowledge in public policy fields. The user can question the conclusions reached by the program and obtain a step-by-step explanation of the inferential process (Han and Kim 1989, 300; Kim, Wiggings, and Wright 1990, 5). Expert systems in public policy and planning are still in their infancy. Nevertheless, in urban planning, for example, several systems have been developed to check compliance with building codes (Heikkila and Blewett 1992), to aid in the site selection process in land use planning, to develop zoning schemes, and to aid in transportation planning and traffic management (Kim, Wiggins, and Wright 1990). These applications have tackled relatively simple tasks.

As the public policy and planning professions tackle more difficult social problems, expert systems may prove inadequate. They are not likely to do well with problems that involve complex and varied inferential processes – problems where different and often conflicting types of considerations, reflecting the multi-disciplinary nature of the problems, play a role. In the artificial intelligence (AI) literature, expert systems are faulted for being "brittle," i.e., subject to failure under dynamic or uncertain conditions. Furthermore, expert systems have not been designed with public users in mind. Although, expert programs are becoming more user-friendly, they have been developed for the use and aid of professionals. It is the professional user, skilled in the expert shell, who benefits from the transparency of the system.

The similarity between my concept of pragmatic knowledge codes and expert systems led me to study developments in AI. Over the past ten years, AI has evolved beyond expert systems. More powerful ways to address machine problem-solving and learning, called architectures, have been developed (Anderson 1983; VanLehn, ed. 1991). Architectures go beyond traditional expert systems by providing theories of cognition as well as programming languages. The expectation is that as architectures evolve, computers will receive less programming and will acquire more knowledge through training and experience. This is similar to the way in which humans acquire expertise.

Since my interest is in developing an organization of knowledge more suited for public education, results cannot be the only concern. The way that results are achieved is also important. My concern extends to the cognitive processes used to arrive at decisions. Ultimately, the organization of knowledge needs to match the knowledge capacity of human beings. Assume, for example, that a traditional expert system could successfully simulate expert decision-making in a particular domain of public policy and planning knowledge. If the goal, however, is to develop intelligent publics for democratic decision-making, then we must still address the issue of the appropriateness of the theory of cognition used. To do this, we have to deal with architectures.

There is a growing number of architectures in various stages of development. Some well-known architectures are ACT* (Anderson 1983), Soar (Laird, Rosenbloom, and Newell 1987; Rosenbloom, Newell, and Laird 1991), Prodigy (Carbonell, Knoblock, and Minton

1991), Theo (Mitchell et al. 1991), and FORR (Epstein 1992a, 1992b, 1994, and 1995). They differ in the methods they use to learn to solve problems, in the number of learning methods used, in the things to be learned and the timing of learning, in the way that explanations are made explicit, in the transparency of what is learned to components within the system, in responsiveness to a dynamic environment, and, of particular interest, in their conflict resolution strategies.

FOr the Right Reasons

Over the past year and a half, I have been working with Susan Epstein, a professor of Computer Science at Hunter College, to determine FORR's (FOr the Right Reasons) applicability to public policy and planning. FORR has great promise for simulating knowledge in public policy and planning. Its potential lies in the two major features that distinguish FORR from the other major architectures mentioned above: its pluralism or reactivity, and its pragmatism.

FORR rejects the assumption that there is a unified reasoning system or agent, in favor of the notion that "the responses from many individual agents can be coordinated reflexively to simulate intelligent decision-making." FORR's "tolerance, even encouragement of discordant good reasons" (Epstein 1994) accomplishes this through a set of Advisors (the good reasons) that aid in decision-making. The capability of accommodating different functional standpoints and social interests, even conflicting ones, is of paramount importance for simulation in public policy and planning domains. FORR's pluralism is also reflected in the multiple ways it can acquire useful knowledge. It can learn, for example, from explanation-based learning, by rote, through induction or deduction. The output from these different learning methods is the basis for the Advisors' comments which influence decisions. The output of the various methods, rather than reliance on a single method, is used to arrive at the right decision based on consensus.

FORR also has a distinctive pragmatic approach toward cognition. It is "an architecture for limited rationality," offering "real-time reasonable behavior, gradual improvement in problem solving, and ability to adapt to a changing environment" (Epstein 1994). While the other architectures rely on the exhaustive and rapid search and extensive memory capabilities of computers, FORR minimizes the use of memory and search. From the standpoint of a knowledge-base for democracy, the requirement is an organization of knowledge and learning that anyone can grasp and internalize, if taught. Such a learning program should not rely extensively on great memory or search capabilities or logical proof, since most persons do not have enormous memories or great conceptual powers. FORR simulates learning and does so with limited memory and search. It thus fulfills an important requirement for a reorganization of knowledge accessible to all persons.

Thus far, Epstein has developed and tested two applications of FORR, Hoyle and Ariadne. Hoyle is a program whose skill domain is two-competitor, perfect information, finite board games. Hoyle has learned to play 18 multicultural games at an expert skill level. The games progress in difficulty from the relatively simple tic-tac-toe to three-dimensional tic-tac-toe, Qubic, Achi, pong hau k'i, tsoro yematatu, and nine men's morris. Although the games do not have the complexity of chess or go, some have various cycle and state transitions. Nine men's morris, for example, the most difficult game Hoyle has mastered, has 14.3 billion nodes in its game tree and an average of 15.5 legal moves for every turn in placing and 7.5 legal moves in sliding. Hoyle avoids extensive forward search into the game tree during play, never looking more than two moves ahead. If there is a conflict among Advisors, Hoyle combines comments to reach a decision. Reaching a decision is accomplished through the organization of Advisors. In Hoyle, Advisors are organized into two tiers. The first tier advisors have absolute authority or veto power. They are consulted in a pre-specified order. The second tier of Advisors, which is not consulted unless the first tier advisors fail to arrive at a decision, are heuristic advisors that can make recommendations in a collaborative way. Decisions in these cases are made by using the fundamental voting paradigm – i.e., tallying the comments and taking the action with the greatest total strength, or a variant of that paradigm (Epstein 1994, 12-13).[4] Compared to other game-playing programs, Hoyle learns more quickly in fewer games and with dramatically less search and memory use.

Ariadne, FORR's most recent application, is a path-finding system: a simulated robot in a rectangular maze (a discrete grid with fixed external walls and internal obstructions), which the robot learns to navigate to a given goal without a map, and with an opportunistic search strategy. Ariadne has performed well on tests in 30 by 30 mazes in a task that is not amenable to traditional AI search techniques. To date, other robot path planning programs have had much less difficult domains than Ariadne (Epstein 1995, 11). As in Hoyle, Ariadne's strength lies in its use of a heuristic set of advisors that enable it to reach a goal without extensive use of memory or planning.

Thalia

Epstein and I have chosen to begin our explorations with the relatively simple knowledge domain of urban forestry. In particular, our work will focus on the selection, planting, and maintenance of street trees. Developing Thalia[5] will require making explicit the useful knowledge and the inferential processes underlying expert decisions in urban forestry.[6]

In specifying the problem class for the simulation, for example, we will have to define what a tree is from the standpoint of our tasks. From the standpoint of planting trees in urban settlements, a tree is something that possesses, among other things, the following characteristics:

- has a trunk with a diameter that increases with age, and a height that varies with species and age;
- has leaves in its branches that vary in size and porosity, depending on species;
- has a canopy of leaves which can be full to sparse, depending on species and seasonal variation;
- has underground roots which require root room in soil, with at least three feet soil depth and ideally a planting plot with a diameter as wide as the tree's canopy.

Such definitions are task-oriented, including only the knowledge that is relevant to the task. The definition above, for example, may leave out facts about the internal structure of trees.

From our discussions so far, it is likely that we will cast Thalia as a board game between two players, Thalia and an outside expert. This will enable Thalia to learn from experience. The board will represent a neighborhood or town of about 50 blocks with some variation in street width, sidewalk width, building heights, traffic conditions, and social conditions (e.g., block organization and the desire for trees). The game will begin with the allocation of a budget (a fraction of funds needed to plant trees in the entire town) for planting trees every few years. The same amount of funds would be allocated to Thalia and to the expert.

One of the tasks will be to determine the location for planting trees in the town, given a limited budget and a set of constraints and opportunities. A set of Advisors, which represent good reasons for planting on a block, will be identified. Such a set of Advisors could include:

NAME	DESCRIPTION	USEFUL KNOWLEDGE
Sunny	Plant on sunny streets	Solar access depends on street orientation, height of buildings, width of street
Demand	Plant on streets where there is greatest demand	Good maintenance is associated with block organization and degree of desire for trees
Roomy	Plant on streets with wide sidewalks	Greatest potential for tree lawns – most desirable urban planting environment
Choking	Plant on streets with most traffic	Potential for pollution mitigation
Sleepy	Plant on streets with least traffic	Potential for longevity

Note that some of the Advisors, Choking and Sleepy for instance, give conflicting advice. FORR addresses such conflict through its organization of advisors. This will be one of the major research tasks we will face in developing Thalia: figuring out which Advisors should have pre-specified authority and which only an advisory role, and how they are to be ranked and sequenced. Since conflict resolution is of such importance to public policy and planning, the choice of using the FORR architecture is especially advantageous. FORR can experiment with voting paradigms "until it finds a reliable form of expertise, i.e., a good way to resolve conflicts among the right reasons" (Epstein 1994). Of course, the FORR architecture could not resolve conflicts among experts, a problem often faced in public policy and planning. But it could model the various ways in which different experts arrive at decisions, including their use of various conflict resolution techniques, and check the models against outcomes.

In a game, a winning goal has to be specified. One such goal could be to maximize the number of trees alive and healthy and their average age after a number of planting cycles. If we also value trees for their ability to absorb CO_2 and other toxins, a winning strategy would have to be more complex. Similarly, if trees are valued both for their shade and their capacity to reduce pollutants in the air, we will have to take into account that trees placed in harm's way (in heavily polluted streets) will have shorter life spans. No matter what, it is clear that we will have to simulate the development of an urban forest over a long period of time, probably 50-100 years, and to include contingencies such as diseases and droughts and fluctuating maintenance budgets.

This discussion provides a sense of some of the tasks involved in developing a planning application in FORR. Overall, it is a formidable conceptual task. Surely, pragmatic knowledge codes for domains of knowledge could be more easily generated on paper than through a cognitive architecture. But there are good reasons for pursuing this task cybernetically. First, developing knowledge codes through a cognitive architecture will help ensure that no relevant knowledge or information will be left implicit. Thus, it fulfills the strong transparency criterion.[7] Second, it is more likely to reveal the processes of learning how to learn, which is as important to a democratic technology as making accessible knowledge domains.

Conclusion

In this article, I have stressed the importance of pragmatic knowledge codes as a way to increase the public's understanding of expert decision-making in policy and planning. From the standpoint of the professions, the development of pragmatic knowledge codes is also desirable. The dilemmas in the general theory of planning set out in Webber and Rittel's (1973, 161 ff.) article on recalcitrant social problems cannot be fully addressed without the development of pragmatic knowledge codes.[8] Without the development of these codes, and without clear and acceptable linkages established among codes, there is no way of knowing which code or problem may be useful, or when the solution strategies in a code no longer help, which code(s) to turn to or sweep in. Without established linkages, any choice among codes remains arbitrary. The development of pragmatic knowledge codes I consider to be a necessary, but not a sufficient condition, for addressing recalcitrant social problems.[9]

Public policy and planning requires interdisciplinary, holistic knowledge of complex systems. We cannot expect that the social sciences will develop such knowledge for us. They are still enmeshed in their one-dimensional quests for hypothetico-deductive systems. We must begin to shoulder this responsibility ourselves.

I suggest that what we need today is a project at the scale of Diderot's and D'Alambert's Encyclopedia, but with a difference. While the Encyclopaedists believed knowledge could be compartmentalized and dispensed in discrete fields and bits – we need to embark on a new knowledge project, one that makes explicit the links and interactions among systems, natural, social, and mixed, and where the focus is as much on the interconnections as on the knowledge codes themselves. Such a knowledge project can set aside ideas of positive science and neutral technology and develop, instead, practical knowledge that is fallible, critical, and evaluative, as well as systems-oriented. What we need is a counter- encyclopedia.

Such a counter-encyclopedia, because of the difficulties in representing procedural knowledge in written form, will be primarily in the form of computer simulations of expert knowledge. Although computer technology is already aggravating existing polarities in access to information and knowledge, it can also prove indispensable in establishing the conditions for genuinely democratic modern societies.

NOTES

[1] Some sections in the first part of this article are condensed versions of the arguments found in my book, *How to think about social problems: American pragmatism and the idea of planning* (1994). The full arguments are contained in Chapters 4 and 7-9.

[2] What do I mean by democracy? Participatory democracy, that is, the participation of average citizens in public policy-making, especially at the local, neighborhood level. With Dewey, I believe that participatory democracy is vital to the formation of strong, integrated personalities, as well as an essential ingredient of a good society. But participatory democracy cannot assume that the average citizen in modern society is "omnicompetent": "competent to frame policies, to judge their results; competent to know in all situations demanding political action what is for his own good, and competent to enforce his idea of good and the will to effect it against contrary forces" (Dewey 1927, 158). Competent and responsible participation in public-policy making requires the formation of intelligent publics. I envisage pragmatic knowledge codes as developing the intelligence needed to create such publics, and thus fulfill a condition necessary to create truly participatory, democratic societies. (See Chapters 4, 6 and 9 of *How to think about social problems* for a full development of this thesis.

[3] This example was initially developed as a class assignment by Lisa Schreibman, a student in my fall 1995 Planning Theory course.

[4] Some variants of the fundamental voting paradigm are:

> smoothed voting, where strengths are converted into "yes" or "no" comments; constrained voting, where only the strongest comment from each Advisor is tallied; constrained, smoothed voting, where only the strongest comment from each Advisor is converted into a "yes" or "no" comment and then tallied. Under all these voting paradigms any ties are broken by random selection. (Epstein 1994, 13)

[5] Thalia was one of the Graces in ancient Greece. The name means, "the flowering."

[6] The tasks of selection, planting, and management of street trees, unlike the example of water quality, are not so much problem-solving tasks as good design or planning tasks. Design/planning tasks are usually more complex than comparable problem-solving tasks, but the bases for good planning decisions can also be made explicit and organized into pragmatic knowledge codes. Instead of simple reliance on diagnostic indicators, planning tasks are constrained by a set of factors to be avoided or optimized, a set of suitability factors that circumscribe choice. In Umberto Eco's terminology, design or planning problems, even within a well-formalized field, would be undercoded tasks, *i.e.*, tasks which could be circumscribed by heuristic rules, but for which there are always a number of possible solutions from which to draw.

[7] Will Thalia be as much of a black box as most computer models? It will be as transparent as expert systems are, that is, transparent to the professional skilled user. This may still mean inaccessible to the lay public. But this is a question of the extent to which expert simulations can be user-friendly, and the answer is that they are becoming more so. Since one of the main reasons I am undertaking this research is to advance the use of these systems by community organizations, and schools, I will do my best to ensure that the inferential process can be accessed in as user-friendly a form as possible.

[8] This other thesis is fully developed in Chapters 7 and 9 of *How to think about social problems*.

[9] The other necessary condition is organizational. (See *How to think about social problems*, Chapters 4, 6, and 9 for the development of this condition).

REFERENCES

Abbott, A. 1988. *The system of professions*. Chicago, IL: University of Chicago Press.

Anderson, J. 1983. *The architecture of cognition*. Cambridge, MA: Harvard University Press.

Blanco, H. 1994. *How to think about social problems: American pragmatism and the idea of planning*. Westport, CT: Greenwood Press.

---. 1993. A systems approach knowledge-base for education. In *Comprehensive systems design*. Edited by C. M. Reigeluth, B. H. Banathy, and J. R. Olson. Heidelberg, Germany: Springer-Verlag.

Carbonell, J. G., C. A. Knoblock, and S. Minton. 1991. Prodigy: An integrated architecture for planning and learning. In *Architectures for intelligence*. Edited by K. VanLehn. Hillsdale, NJ: Lawrence Erlbaum.

Commoner, Barry. 1987. A reporter at large: The environment. *The New Yorker*, 15 June, 46-71.

Dewey, John. 1927. *The public and its problems*. New York City, NY: H. Holt.

Eco, Umberto. 1979. *A theory of semiotics*. Bloomington, IN: Indiana University Press.

Edmunds, R. A. 1988. *The Prentice-Hall guide to expert systems*. Englewood Cliffs, NJ: Prentice-Hall.

Epstein, S. L. 1992a. Prior knowledge strengthens learning to control in weak theory domains. *International Journal of Intelligent Systems* 7: 547-586.

---. 1992b. The role of memory and concepts in learning. *Minds and Machines* 2: 239-265.

---. 1994. For the Right Reasons: The FORR architecture for learning in a skill domain. *Cognitive Science* 18: 479-511.

---. 1995. On heuristic reasoning, reactivity, and search. In *IJCAI-95: Proceedings of the Fourteenth International Joint Conference on Artificial Intelligence*, 20-25 August, in Montréal, Québec, Canada. pp. 454-61. San Mateo, CA: IJCAI, distributed by Morgan Kaufmann Publishers.

Gupta, A., and B. Prasad, eds. 1988. *Microcomputer-based expert systems*. New York City, NY: IEEE Press.

Han, S., and T. J. Kim. 1989. Can expert systems help with planning? *Journal of the American Planning Association* 55: 296-308.

Heikkila, E. J., and E. J. Blewett. 1992. Using expert systems to check compliance with municipal building codes. *Journal of the American Planning Association* 58(1): 72-80.

Kaiser, E. J., D. R. Godschalk, and F. S. Chapin, Jr. 1995. *Urban land use planning*. 4th ed. Urbana and Chicago, IL: University of Illinois Press.

Kim, T. J., L. L. Wiggins, and J. R. Wright, eds. 1990. *Expert systems: Applications to urban planning*. New York City, NY: Springer-Verlag.

Laird, J. E., P. E. Rosenbloom, and A. Newell. 1987. SOAR: An architecture for general intelligence. *Artificial Intelligence* 33(1): 1-64.

Michie, D., ed. 1982. *Introductory readings in expert systems*. New York City, NY: Gordon and Breach.

Mitchell, T. M., *et al*. 1991. Theo: A framework for self-improving systems. In *Architectures for intelligence*. Edited by K. VanLehn. Hillsdale, NJ: Lawrence Erlbaum.

Rittel, Horst, and Mel Webber. 1973. Dilemmas in a general theory of planning. *Policy Sciences* 4: 155-69.

Rosenbloom, P., A. Newell, and J. E. Laird. 1991. Toward the knowledge level in SOAR: The role of architecture in the use of knowledge. In *Architectures for intelligence*. Edited by K. VanLehn. Hillsdale, NJ: Lawrence Erlbaum.

VanLehn, K., ed. 1991. *Architectures for intelligence*. Hillsdale, NJ: Lawrence Erlbaum.