Indexing Evaluations of Buildings to Aid Conceptual Design *

Anna L. Zacherl and Eric A. Domeshek
College Of Computing
Georgia Institute of Technology
Atlanta, GA 30332-0280
{anna or domeshek}@cc.gatech.edu

Abstract

The cognitive model underlying Case-Based Reasoning (CBR) has implications for human performance on many tasks, and the technology developed in CBR research can be turned to enhancing performance. For all case-based systems, effective memory retrieval depends on a successful assault on the indexing problem. But demands on indexes may be different, and in some ways, more stringent, for aiding systems than for autonomous problem-solving systems.

This paper reports on the evolution of an indexing scheme intended to retrieve lessons for architects working on conceptual design of buildings. It illustrates, by example, the process of designing indexing systems, and the demands peculiar to indexes for aiding systems.

Finding Experiences for Experts

It is the norm for expert performance in cognitively challenging tasks to depend on extensive experience. Researchers in the AI paradigm of Case-Based Reasoning (CBR) have been building computational models that account for this fact, and have aimed to produce systems that perform effectively by relying on records of past experiences [Hammond, 1989; Hinrichs, 1992; Koton, 1988; Mark, 1989].

More recently, insights gleaned from a decade of CBR research have been turned towards the problem of building systems that aid humans in performance of real-world tasks [Schank *et al.*, 19891; Ferguson *et al.*, 1991]. At Georgia Tech, we have been focusing on ***design tasks*** such as architecture, engineering, and lesson planning [Domeshek and Kolodner, 1992b; Domeshek and Kolodner, 1992a; Chandler and Kolodner, 1993]. We are attempting to produce what we call *Case-Based Design Aids* (CBDAs) – systems that help human designers by making available to them a broad range of critiqued designs that can serve to highlight important design issues, to explicate abstract design guidelines, and to provide suggestions or warnings about possible design solutions.

Any system that bases its performance on the selective use of items from a large memory must find some way to organize that memory so that the right items can be found at the right time. In CBR, this problem of how to ensure effective selective retrieval goes by the name of ***the indexing problem***, and has long been recognized as one of the key issues in the field [Schank, 1982; Hammond, 1989; Domeshek, 1992]. Research has provided some insight into how this challenge must be addressed, and has produced a sampling of exemplary indexing systems for particular domains and tasks.

This paper reports on the development of an indexing system for one of our CBDAs – **Archie-II** an aid for conceptual design in architecture. The story we have to tell of the evolution of this indexing system is interesting for several reasons. First of all, there has been relatively little work in the CBR community on indexing systems for physical artifact design tasks. Secondly, because we have been designing indexes for an *aiding* system we have had the burden and the opportunity to grapple with cases of far greater complexity than is possible when working on autonomous reasoners; the domains in which it is practical to build autonomous systems are necessarily much less complex than those in which humans routinely engage (and in which humans are likely to need aid). Finally, it is an interesting question to what extent the demands of indexing for autonomous and aiding systems may differ; the information available in situation descriptions (which is thus information easily available for indexing

*Many people have worked on the development and conception of the system described in this paper. Janet Kolodner and Craig Zimring have shaped this project from the start. Richard Billington, our research programmer, has been a constant collaborator throughout. The **Archie-II** team also includes Kadayam Vijaya, Ali Malkawi, Ellen Do, and David Brogan. Interviews with architects have been invaluable; the architects included Lane Duncan, Rufus Hughes, Michel Lincolt, and Von Rivers. This work has been supported in part by the Defense Advanced Research Projects Agency, monitored by ONR under contract N00014-91-J-4092. All views expressed are those of the authors.

to past experiences), may be different when a system is engaged in problem solving than when it is responding to a user's queries.[1]

In the context of an aiding system, the indexing problem shades into an HCI or interface problem, and in the face of a usefully complex domain such as conceptual design of buildings, the usability problem is exacerbated. What is required to succeed here is an understanding of the processes used by our users: how do architects think about their task during the early stages of design?

Indexes to Design Lessons

The first thing to note when considering how to index design cases for architects is that *buildings* are probably not the right units of storage and retrieval. Buildings are too big and complex to be designed as one piece; instead there are many small decisions that go into a building design, and what would best serve an architect are lessons about which issues are important, how to address these important problems, and what outcomes are likely. Thus, the primary unit of memory in **Archie-II** is the lesson-bearing *story*. As an example, consider the following story [Building Diagnostics, 1988]:

> *The location of the main lobby information desk in the Bristol County Courthouse is inconvenient and makes it difficult for people to find their way. The desk is located in the telephone office, which is off to the left of the main lobby entry. There is a small sign on the telephone office indicating that it is also the information desk, but on first entering the courthouse there is no immediate indication where to go for information.*[2]

Since the items being indexed are stories that teach lessons about design issues, it is appropriate that indexes to these stories center on design issues. But issues vary from one part of a building to another, and even pervasive issues, such as efficient circulation may vary significantly in their implications throughout the building. Accordingly, our stories tend to focus on how an issue plays out in some part of a building, and thus our indexes must also specify the relevant parts. Issues also tend to arise at different times during the life of a building: some are important during construction, others during use or renovation. Likewise, not all issues affect all of the stakeholders in a building: some are of most importance to the owners, others to long term residents, and still others to occasional visitors.

With one final distinction, this analysis of features that differentiate issues provides an outline for our indexes. We recognize two different ways of slicing a building into parts: spatially and functionally. Spaces are physically localized building chunks such as floors, wings, or offices. Functional systems such as the electrical and plumbing systems may be distributed throughout a building and are defined in terms of their purpose. So we recognize five primary dimensions as relevant to describing the point of a story with a simple lesson:

- **Issue:** Goal to be achieved by the artifact's design
- **Space:** Part of designed artifact defined spatially
- **System:** Part of designed artifact defined functionally
- **Stakeholder:** Role with respect to artifact defining a point of view on the issue
- **Life Cycle:** Part of artifact's history when the issue matters

A combination of some subset of these descriptors is sufficient to identify a single point that might characterize the lesson of one of our stories (thus serving as a memory *label*), or might express a user's browsing interests (thus serving as a retrieval *probe*). But often, a story is interesting because of what it says about the *interaction* between issues. In one courthouse, the prisoner holding area was located far from the courtrooms, which led to a desirable lack of noise in the court, but also contributed to security problems when the prisoners were being transferred through the building. Stories that address the interaction between issues are best indexed by a pair of the five-featured structures above.

The index outline just sketched, whether used singly or in pairs, does not say much about what sorts of issues, spaces, systems and so forth we will have to represent. It is essentially a road map to the work required for a fully specified indexing system. To give a sense for the how such a system is developed, this paper will concentrate on just one of these dimensions: *spaces*.

Designing an Indexing Vocabulary

Designing an indexing vocabulary is an exercise in exploring the possible descriptions of objects, concepts, and relationships in the domain, and settling on a system that meets several criteria [Kolodner, 1993]:

1. **Relevance:** Index vocabulary must capture those aspects of situations that indicate when one is relevant to another (with respect to a task or tasks). This is just a baseline, common sense guide to help decide what sorts of features ought to be included in an indexing vocabulary.
2. **Extent:** Index vocabulary must be sufficiently extensive to describe an existing or expected corpus of

[1] An aiding system looking over a user's shoulder, and trying to explain aspects of the user's problem solving process, likely results in still different sorts of information being easily available for indexing.

[2] Note that **Archie-II**'s stories are always shown with a presentation (usually graphic) of the artifact being discussed, and are generally accompanied by an illustration that amplifies the point being made in the text.

memory items for the range of intended uses. We want to cover a large corpus of design stories, and eventually would like to allow users to enter their own stories.

3. **Specificity:** Index vocabulary must get specific enough to make all useful discriminations among items in memory. We want to retrieve only those stories that are most relevant to a designer's situation.
4. **Generality:** Index vocabulary must also contain components general enough to capture relevant *similarities* among the items in memory. We want to be able to retrieve stories whose indexes are inexact matches to a designer's situation, if that is the best there is in memory.
5. **Usability:** In an aiding system, it is helpful if the indexing vocabulary corresponds closely to *practitioners' conceptions* of their domain and task. We want designers to feel comfortable using the system and we want to minimize the amount of inference required of the system at retrieval time.

Actually, extent, specificity and generality are all closely related. What it means to cover a corpus (to have sufficient extent) is just to be able to note the similarities and differences between the items in the corpus. Any (relevant) feature introduced into an indexing vocabulary is likely to improve the system's extent, and may serve both specificity and generality; the feature will appear in some indexes but not in others, and thus will discriminate; but for all indexes in which the feature appears, it will be capturing a similarity.

An architectural design aiding system presents serious challenges in balancing the criteria. To be useful, we must build a large corpus, we must index it using terms an architect might naturally employ, and we must build in sufficient flexibility to satisfy the often idiosyncratic approaches of many different architects. All this must be accomplished while not burdening the user (or the indexer) with too many choices.

Satisfying all the criteria in any interesting domain is actually a hard design problem. Arriving at an acceptable solution generally requires an iterative process of *analyzing* the domain and task, *proposing* index components, and *evaluating* those components with respect to the criteria (which may involve actually *building* partial memories based on those proposals). We have been following this process for **Archie-II**'s indexes, and are currently on the third major loop through the cycle for the space components. The next section describes the history of that design process with particular attention to the rationale for our current solution.

Spaces as Index Components

The possible spaces in a building are many and various. There are also many possible ways of *describing* those spaces. Here we summarize three approaches considered for the **Archie-II** system.

Spaces: A First Pass

The first attempt at an indexing vocabulary for spaces was developed, in part, to suggest the way such attempts might proceed in general. Based on a corpus of stories we then had in hand, a list of quite specific space types was created; since our cases were drawn from courthouses, this list included such items as parking-lot, entry, lobby, information-desk, vestibule, and courtroom. Just as we characterized our initial index proposal in terms of a set of five dimensions, we proceeded to pick out dimensions (in this case four of them) that together began to characterize the spaces in our list.

The four dimensions selected to describe spaces were ownership, purpose, size, and position. *Ownership* encoded common patterns in who used the space. *Purpose* was intended to indicate what the space was used for. *Size* was specified in terms of square footage. Finally, *position* encoded the space's location within the building.

For each dimension we then proceeded to specify a set of possible fillers. In principle, this system was not limited to the original list of named spaces – any combination of the defined fillers for any subset of the dimensions could be used to describe a space. In practice, these dimensions and the fillers provided for them were not even sufficient to do a good job of describing the spaces in our original list. For instance, we could not distinguish criminal from civil courtrooms.

Still, this initial proposal had some positive features. Because it allowed use of a set of everyday names for spaces, it was relatively easy for an architect to use. The interface problem created by a potentially lengthy list of names to choose from was somewhat mitigated by the ability to enter a partial description of a space (by choosing fillers for any of the dimensions); given such a partial description, the system would then prompt the user with a limited menu, including only those spaces that satisfied the specified conditions. Of course, here the limited expressivity of the space description language became a problem, as did the fact that choosing fillers for the dimensions was not as intuitive as the straightforward choice of space names.

Limiting the characterization of spaces to the four dimensions and their fillers not only affected usability, but also failed the tests of extent, specificity and generality. The following example illustrates a failure of specificity. Imagine an architect is designing an office suite for a large accounting department and is concerned with appropriate use of natural lighting. Currently, **Archie-II** only stores courthouse stories, so while it has stories about lighting in several office spaces, the names associated with the spaces include "probation office", "judge's lobby", and "magistrate's office", but not "accounting department". Now, by choosing from among available values for the purpose and ownership dimensions the user could tell the system that the space she is concerned with is a work area generally off limits to the public. On that basis, the system might retrieve stories associated with any of the courthouse office spaces mentioned earlier. But consider that of all those spaces, there is one that is clearly most similar to an accounting department; the probation office is an office suite that includes a large private office for the manager, a group of smaller offices for staff, and a place for a receptionist. Other courthouse office spaces do not house the same number of people, nor break up the space with the same sorts of dividers. When the system retrieves lighting stories from all the courthouse offices, the user is forced to sift through a large collection of stories, most of which are unlikely to be relevant to her lighting problem.

Spaces: A Second Pass

In our second attempt at a vocabulary for spaces our primary concerns were to increase the expressivity of the language and to take better account of the features that mattered during conceptual design (and thus that would determine the relevance of a story's lesson). Among the constraints architects face when designing spaces, some of the most powerful stem from *who will use the space, what they will use it for, and what kinds of support are required for such use.* For example, a room where the primary activity is discussion must be set up differently than a room where the primary activity is lecturing; a room for discussions by groups of 20 must be designed differently than a room for discussions by 3 or 4; a typical manager's office must function not only as a small group discussion room, but also as a place for private desk-work.

In a sense, answering these questions would force us to expand on the original dimensions of ownership and purpose. A separate effort was anticipated to capture more of the physical attributes of spaces (in effect, expanding beyond the original pair of dimensions, size and position); note that this second extension would more likely address the shortcomings identified in the office lighting example of the last section. The focus on issues critical to conceptual design also led us to introduce in this second pass an important set of features that had not been considered at all the first time around. Often a space's design is strongly influenced by its interactions with other spaces. So while focusing on uses of spaces, we also began to look at the *relationships* between uses of separate spaces.

As a way of encoding descriptions of the people, activities, and props associated with a space we adopted the *script* formalism [Schank and Abelson, 1977]. Under this proposal, the space dimension of our five-part index outline was to be filled with a set of scripts and references to related spaces. Similarities among the specific activities, role-fillers, props, and spaces would help determine the system's judgments of similarity between story indexes and user queries.

Below is an example of a script-based space description for a court clinic (a court clinic is where a psychologist or social worker counsels young probationers):

- **Court Clinic:**
  - **Role:**
    - Psychologist / social worker
    - Juvenile probationer
  - **Props:**
    - Desk
    - Toys / toy storage
  - **Activities:**
    - Psychologist talks to probationer
    - Probationer talks to psychologist
    - Psychologist works at desk
    - Probationer plays with toys
  - **Related Spaces:**
    - Waiting room
    - Juvenile probation office
    - Conference room

The kinds of information included in this space description are, in fact, important to architects; our second pass at index design improves on the original scheme by significantly broadening its *extent.* Unfortunately, this approach swings too far in the direction of detail. Users will not put up with having to constantly be explicit about much that is normally left as tacit knowledge about a design problem, so *usability* has actually deteriorated. What makes this more than just an interface problem is that there still remain important similarities and differences among spaces that cannot be described in terms of scripts and related spaces (or in terms of some improved vocabulary for physical description).

Consider the constraints on a juvenile courtroom. A description of what goes on in a juvenile trial will not really be able to capture the important notion of *confidentiality,* and a design for a juvenile courtroom will not succeed without taking confidentiality into account. The fact that court proceedings are supposed to remain confidential is actually a matter of what does not happen: someone not associated with the trial does not get to observe it. Scripts do not normally include such negative statements, and even if they did, requiring a user to express such a basic concept in such particular terms would be awkward.

Finally, we note that in addition to excluding some important features, the scriptal approach to describing spaces may include many features that do not contribute much to discriminating stories in our corpus. For example, while the need for props such as a desk, file storage, chairs, and tables in a space for office-work places some constraint on the space design, when dealing with a building like a courthouse, where many of the spaces share that feature, we are not getting much return for our descriptive effort.

Spaces: A Third Pass

The first attempt at an indexing vocabulary for spaces captured only a small number of the relevant features, and the second attempt, while capturing more, still neglected many important abstractions and required too much attention to detail. Our strategy for arriving at a system with the right amount and level of detail was to go back and pay more attention to the way architects do the work we aim to support.

Accordingly, we have been devoting more effort to studying architects' processes of conceptual design. We need answers to the pair of questions: What kinds of decisions do architects make during conceptual design, and what features of the design problem and its evolving solution do they use to make those decisions? So far, we have a preliminary answer to the first question that breaks down conceptual design issues into two major categories: the **organization** of spaces and the features of individual spaces. The features of the spaces break down, in turn into three categories: the people and things that play important **roles** in the space, features of the **exterior**, and features of the **interior**. So in all, we have four clusters of features to consider.

There are four components to space **organization**:

- O1) Relationships between spaces
- O2) The strength of inter-space relationships
- O3) Distances between spaces
- O4) The orientation of spaces to the site

Roles fit into three categories. *Primary* and *secondary* roles indicate who is using a space, differentiating between those who use the space frequently and those using it less frequently. *Props* are inanimate objects associated with particular spaces. These then are the three role components:

- R1) Primary roles
- R2) Secondary roles
- R3) Props

The **exterior** can be described along four dimensions:

- E1) Openings (such as doorways and windows)
- E2) Materials
- E3) Three-dimensional shape or form
- E4) Space flow (paths between spaces, and the 3-D shape of those paths)

Finally, **interiors** can be described using three dimensions:

- I1) Function
- I2) Materials
- I3) Space characteristics (such as size, lighting, and thermal comfort)

This outline of conceptual design issues serves as the basis for space descriptions in our indexing vocabulary. Most of these features apply to a space; some apply to *relationships* between a space and some other space. When describing any particular space (considered the *focal* space) we characterize it using both sets of features. The features that characterize the space's relationships to other spaces implicate two other sets of spaces: *included* spaces cover parts of the focal space (differentiated based on function); *related* spaces are other disjoint spaces with interesting relationships to the focal space. The included and related spaces can in turn be described using these kinds of features. Applying these features to a focal space and its relationships to other spaces results in the following form for space descriptions:

- **Focal Space**
  - O4 - Orientation to site
  - R1 - Primary Roles
  - R2 - Secondary Roles
  - R3 - Props
  - E1 - Exterior openings
  - E2 - Exterior materials
  - E3 - Form
  - I1 - Function
  - I2 - Interior material
  - I3 - Space characteristics
  - **Included Spaces** [Pointers to other spaces]
  - **Related Spaces** [Pointers to other spaces]
    - O1 - Relationship type
    - O2 - Relationship strength
    - O3 - Distance
    - E4 - Spaceflow

Most features apply directly to a particular space, and are shown here associated with the focal space. Note that included and related spaces are themselves spaces that can, potentially, be described by the same features. A smaller set of features (O1-O3 and E4) bear directly on relationships between spaces, and these are shown here nested beneath the related spaces.

At any given point in the design process, the architect is most directly concerned with some particular level of analysis which defines what count as focal spaces and what count as included spaces. For example, during conceptual design of an entire courthouse, a likely unit of analysis is the judges' lobby, consisting of judges' offices plus associated support areas such as clerks' offices and private restrooms. At the point where the judges' lobby is a focal space, we expect most of the relationships of interest to be expressed as relationships between the lobby and other areas, such as the courtroom. This contrasts with another, finer level of analysis, at which relationships might be noted between included spaces such as a judge's office and the judge's bench in the courtroom.

We illustrate this new approach by describing a jury room as part of a query about juror circulation patterns to and from the room. The architect knows that jurors play a primary role not just in their jury deliberation room, but also in courtrooms and in the jury pool room (where jurors are selected for trials). If the architect is concerned about how to arrange circulation to and from a jury room, the following description of that room can serve as a useful piece of a query for relevant stories:[3]

Focal Space	Jury Room
R1 Primary Roles	Jurors
I1 Function	Isolate primary role
I3 Space characteristics	
Acoustics	Isolation
Included Space	Meeting Area
Included Space	Private Restroom
Related Space	Jury Pool Room
O1 Relationship type	Same primary role
O2 Relationship strength	5
Related Space	Courtroom
O1 Relationship type	Same primary role
O2 Relationship strength	5

In this description the jury deliberation room is the focal space with respect to which the architect is concerned about circulation. The architect specifies only a few aspects of this room: its primary residents will be jurors, and its primary function is to keep jury deliberations confidential (which is to be accomplished by physically isolating the jurors from other building users and by making sure that the room is acoustically isolated from other spaces). The two included spaces describe a logical guess by the architect: there should be room to allow the jurors to discuss the case, and there should be a private restroom. In this context, the architect specifies no further details of the included spaces (which could, potentially, have been treated as fully specified spaces in their own rights). The two related spaces are mentioned because the architect sees their overlapping primary roles as important; her purpose is to explore the implications of these relationships. The relationships' types are characterized in terms of the shared role fillers, and their strengths are rated very high - they are pegged at 5 on a scale from 1 to 5.

The space description above would be accompanied by other index components to indicate the architect's interest in stories about the implications of jurors' circulation to and from the Jury Room during normal use of the building for trials. Given this description, the system could have a basis for choosing the following warning story describing circulation patterns for jurors moving between the related spaces that fail to ensure desired isolation:

> *On the second floor, the superior court and municipal court jury assembly rooms are placed around a service core that serves both of them. This core contains direct access to the stairs and elevator used by the staff. This can give jurors an opportunity to take the elevator to other floors and make contact with people in the staff areas, which are supposedly segregated from the public.*

This system, based on modeling the indexes after the actual decisions made during conceptual design, does better at satisfying our criteria than did the previous attempts. Extent, specificity and generality are all improved by attending to a wider range of features that are attended to by architects. This scheme is not as susceptible to the sort of failing we saw with our first proposal, when the system could not notice how an "accounting department" was more like a "probation office" than a "judges' lobby" or "magistrate's office." Despite including descriptors such as the role filler "juror" this scheme also need not be unduely bound to the idiosyncracies of the courthouse domain; we have preliminary breakdowns of such categories in terms of underlying attributes that capture their relationship to the building and its spaces. We take it as a constraint on the vocabulary items we posit that they not only contribute to distinguishing among our courthouse cases, but that they also contribute to identifying distinctions likely to matter in other architectural (and even other design) domains.

Bubble Diagrams and a Bubble Editor

Hewing to architects' own distinctions and vocabulary as done in the third pass, should improve expressiveness. Unfortunately, the complexity of this query format is likely to raise serious problems with usability. We are therefore trying to provide a reasonably intuitive way for architects to designate spaces of interest. Architects love graphic representations, and are adept at visualizing spaces. It would be nice, if we could allow our users to pose queries simply by pointing at representations of spaces on screen. The problem is that there is no canonical graphic representation for all the features of spaces relevant during conceptual design (and, it would premature to draw detailed CAD diagrams at such an early stage of design).

It turns out that architects have, however, developed graphic forms appropriate to conceptual design: many architects develop their early ideas using *bubble diagrams*. In a bubble diagram, a bubble (a blob that does not necessarily represent shape but does represent size) is drawn to represent each space. Lines are drawn between bubbles to indicate relationships: dark, thick lines represent strong relationships while light, thin lines represent weaker relationships.

We intend to provide users of **Archie-II** with a bubble diagram editor for creating and viewing these

[3] Note that many attributes are left blank because the architect is not yet committed to their values, and retrieved stories may offer advice on how to flesh them out.

schematic design aids. Ideally this tool will comfortably support their normal conventions but will also make it easier to capture the features we need for query processing. When a bubble is drawn in the editor a template will pop up on the screen asking the user to fill in the blanks for the various features of the space. When a line is drawn between bubbles another template will pop up asking the user to describe the relationship between the two spaces.

The bubble editor will assist the architect with conceptual design in three ways. The first way is simply to improve on a representational technique they are already using by automating editing and preserving an on-line record of their work. The second way is by helping the architect describe features of the spaces. The third way is by helping to form queries to the case-based design aiding system.

Conclusion and Future Work

This paper has focused on two issues. The first was explaining and illustrating the iterative process of designing an indexing system. The second was arguing for the appropriateness of the current system for indexing stories that teach lessons about conceptual design in architecture. By describing stories using features that architects consider during conceptual design, the indexing system should naturally provide users with an easy way to specify probes to the case-based system.

It is interesting to study both the evolution of our indexing system and its current state. Everything seemed to fall into place once we focused on basing the index vocabulary on the decisions made during conceptual design. This made the vocabulary familiar to the architect and opened up opportunities for graphical representations. It seems like common sense to base the vocabulary on the decisions made, but it is sometimes useful to have the obvious stated. Although we have not made the attempt yet, we believe that this approach is transferable to conceptual design in other fields.

Further studies with architects, including interviews and observation, are planned to help us refine this approach still further. We hope to learn more about what space characteristics are important to architects during conceptual design. We also hope to learn more about how architects use bubble diagrams, since they appear to be a strategically impoverished representation that force focus on those central issues that matter early in design.

References

Building Diagnostics, Inc. 1988. Post-occupancy evaluation: New bedford county courthouse. Post-Occupancy Evaluation CBR85-1STU, Office of Programming, Boston, MA. Division of Capital Planning and Operations.

Chandler, T.N. and Kolodner, J.L. 1993. The science education advisor: A case-based advising system for lesson planning.

Domeshek, E.A. and Kolodner, J.L. 1992a. A case-based design aid for architecture. In Gero, J.S., editor 1992a, *Artificial Intelligence in Design 92.* Kluwer Academic Publishers, The Netherlands. 497–516.

Domeshek, E.A. and Kolodner, J.L. 1992b. Towards a case-based aid for conceptual design. *International Journal of Expert Systems* 4(2):201–220.

Domeshek, E.A. 1992. *Do the Right Things: A Component Theory for Indexing Stories as Social Advice.* Ph.D. Dissertation, The Institute for the Learning Sciences, Northwestern University.

Ferguson, W.; Bareiss, R.; Osgood, R.; and Birnbaum, L. 1991. Ask systems: An approach to story-based teaching. In *Proceedings of the International Conference on the Learning Sciences*, Charlottesville, VA.

Hammond, K.J. 1989. *Case-based Planning: Viewing Planning as a Memory Task.* Academic Press, Boston, MA.

Hinrichs, T.R. 1992. *Problem Solving in Open Worlds: A Case Study in Design.* Lawrence Erlbaum Associates, Hillsdale, NJ.

Kolodner, J.L. 1993. *Case-Based Reasoning.* Morgan Kaufmann, Los Altos, CA.

Koton, P. 1988. Integrating case-based and causal reasoning. In *Proceedings of the Tenth Annual Conference of the Cognitive Science Society.*

Mark, W.S. 1989. Case-based reasoning for autoclave management. In *Proceedings of the Second Case-based Reasoning Workshop*, Los Altos, CA.

Schank, R. and Abelson, R. 1977. *Scripts Plans Goals and Understanding: An Inquiry into Human Knowledge Structures.* Lawrence Erlbaum Associates, Hillsdale, NJ.

Schank, R.; Ferguson, W.; Birnbaum, L.; Barger, J.; and Greising, M. 19891. Ask tom: An experimental interface for video case libraries. Computer Science Technical Report 10, Institute for the Learning Sciences, Northwestern University.

Schank, R. 1982. *Dynamic Memory: A theory of reminding and learning in computers and people.* Cambridge University Press, Cambridge, UK.