Perspectives in Explanation

Daniel D. Suthers [1]

COINS Technical Report 89-24

Department of Computer and Information Science
University of Massachusetts
Amherst, Massachusetts 01003

suthers@cs.umass.edu

Abstract — This report reviews artificial intelligence research relevant to computational theories of explanation as a communicative activity. The literature is organized into three broad areas: epistemological adequacy, i.e. providing the diversity of knowledge needed in explanations; epistemological and rhetorical analyses of explanatory structure; and alternate mechanisms for generating explanations. Several research problems and design issues are emphasized. Two problems occur at an epistemological level of analysis. There is scattered research accounting for how explanations are constrained by the internal structure of a domain's body of knowledge, by the role of an individual's knowledge in understanding new concepts and situations, and by the ways in which individuals are willing or able to transform their knowledge. These epistemological constraints on explanation require more explicit treatment within a uniform framework. Further work is required to identify and represent alternate conceptual frameworks or "perspectives" taken in an explanation, and to account for an explainer's choice of a good perspective for a given explanatory goal. A third problem is the design of a framework within which to operationalize the various constraints and heuristics defining a "good" explanation and coordinate and integrate them with each other. A suggestion is made, in terms of the sources of these constraints and their impact on a process of translation between knowledge structures. Finally, several secondary issues discussed are concerned with the expression and implementation of a theory of explanation, including the utility of various abstractions and computational devices, and the extent to which explanation facilities should be coupled to other aspects of the design and operation of a knowledge-based system.

 Prepared for the Ph.D. qualifying examination in Computer Science at the University of Massachusetts. The author has been supported by the National Science Foundation under grant number MDR 8751362, and by Apple Computer, Inc., Cupertino, CA. Partial support was also received from the Office of Naval Research under a University Research Initiative Grant, contract no. N00014-86-K-0764.

Contents

List of Figures

Acknowledgments. Thanks to Edwina Rissland and Bev Woolf for advising; Sara Betz, Jamie Callan, Adele Howe, Philip Johnson, Bob Krovetz, Victor Lesser, David Lewis, Frank Linton, Tom Murray, Ted Norton, Tony Priest, Klaus Schultz, Kishore Swaminathan, and Paul Utgoff for discussion and feedback of various sorts; Sharon Rose and Geoffrey Suthers for tolerating obsessive behavior; and Joe Sullivan for encouraging me to articulate my perspective on my chosen field before getting lost in the details of my dissertation.

Chapter 1

Aims and Issues

This document reviews Artificial Intelligence (AI) research relevant to supporting explanation in knowledge-based systems.[1] While a variety of material has been included for the sake of an accurate assessment of the state of the art, some dominant themes have emerged. These have to do with choice of perspective and structuring the content of explanations to make contact with and build on the hearer's knowledge; identification and organization of the constraints which bear on these choices; and secondary issues in expressing and implementing a theory of explanation. Explanation is a centering point for a group of related issues in the study of "knowledge communication", a paradigm which I briefly examine before previewing the issues and organization of this paper.

1.1 Knowledge Communication and Explanation

Wenger (1987) presents **knowledge communication** as an integral part of intelligence, on a par with problem solving or learning. He writes:

> "... knowledge communication is defined as the *ability to cause and/or support the acquisition of one's knowledge by someone else, via a restricted set of communication operations.* ... I have abstained from attempting to define knowledge or communication. Such definitions would stand in contradiction to the claim that the enterprise described here is intrinsically bound to an inquiry into the nature of these concepts." (p. 7)

I hesitate to define explanation at this juncture for similar reasons. A precise definition would assume completion of this line of research. By the end of this review, I will have collected enough material to place constraints on what constitutes a good explanation. Meanwhile, some informal comments will provide the reader with a feel for what is included. For the purposes of this review, **explanation**

[1]The definition of "explanation" will be discussed. Knowledge-based systems are computer programs that rely primarily on representations of (large amounts of) domain knowledge to perform their task. Usually the knowledge is represented in an explicit form accessed by one or more interpreters which consult these representations to determine what the program will do next. Knowledge-based systems have also been characterised as relying on universally quantified statements in addition to ground facts (as in data base systems), and their reasoning is heuristic as well as algorithmic. "Artificial Intelligence" defies principled definitions, and is perhaps best understood as a subculture of computer science whose members are attempting to model human mental and perceptual abilities with computational theories, pushing the boundaries of what machines can do.

is to be taken in its broad sense which includes descriptions of objects, processes, or knowledge, as well as justification of reasoning, actions, or beliefs. It is, as the American Heritage Dictionary tells us, "the act or process of making plain or comprehensible; elucidation; clarification". **Explanation research** is clearly a subfield of the study of knowledge communication. It is a *proper* subfield in that other aspects of knowledge communication are de-emphasized, though not necessarily neglected outright. Indirect means of communication, such as getting an interlocutor[2] to discover things through experimentation or Socratic dialogue techniques, are of lesser concern. Diagnostic activities, such as the construction of a user model, fall outside its realm, though a theory of explanation should specify how to use the resulting information. Put concisely, explanation research addresses how the content and structure of explicit communications, directed towards specific informative goals, are generated under a variety of constraints to be discussed in this review.

I am particularly oriented towards computational theories of explanation by which computers may communicate scientific and technical knowledge and its use. **Knowledge communication systems** include those knowledge-based systems which are intended for interactive use, such as advisory systems which aid users in dealing with some problem requiring the application of knowledge not normally available to the targeted users, and tutoring systems which are designed to teach in a manner sensitive to the individual. In my view, knowledge communication systems constitute a fundamentally different class of applications of computers that require new methodologies in their design. The use of computers for knowledge communication may be distinguished by comparison to conventional computer applications on the one hand, and existing media for knowledge communication on the other.

In conventional applications, the users often come to the machine having already evaluated and chosen a given approach (algorithm, statistical test, data base model, etc.). The point of the implementation is to get it to retrieve the right data or give correct results for complex computations. These results are then interpreted by the user based on knowledge he or she already has about the problem domain. In many knowledge-based systems, an additional aspect of the machine's task is to communicate to the users the knowledge which will enable them to use the results of the system. This may require defining terms, explaining how evidence is used, identifying assumptions, describing the domain theory behind the reasoning, and providing whatever other information is needed to evaluate the correctness or applicability of its results. The machine should determine when such communicative acts are appropriate, and attempt to tailor its actions to the knowledge and goals of the user. Thus, the step from conventional applications to knowledge communication systems requires research in designing rich representations of domain knowledge, and making knowledge about domain knowledge and how to communicate it explicit and usable according to a computational theory. The explanation research reviewed here attempts to make this contribution.[3]

Written and printed pictorial communication media typically are structured according to some schematic form (eg. table of contents, introduction, chapters with various headings and subdivisions, summary, index and references). While this structure can be very rich, the user is responsible for utilizing the indexing methods. Indices exist at a coarse granularity, and hence their use is mixed with sequential search. The material, once created, is fixed in its presentation style, including emphasis, choice of conceptual viewpoint on the material, and terminology used. Video and film media provide

[2]This fancy term is sometimes more appropriate than "user", "student", or "hearer". I use it when the discussion is not restricted to users of a computer program or to a pedagogical context, or when the role of the other person as an active dialogue participant is to be emphasised.

[3]Hill (1989) argues that all of Artificial Intelligence has as its most basic aim the design of computation-based representation media. If so, this contribution is not limited to explanation research.

a rich perceptual experience unequalled elsewhere, but are usually accessed sequentially and are also fixed in their presentation. Computer-based media are already surpassing the richness of structure afforded by written material (witness hypertext), and can borrow the visual richness of video forms of presentation via mixed-media hardware. In addition, they have a unique potential for *dynamic, context sensitive organization and selection of material.* The form of the material to be presented need not be fixed when the work is "completed". The conditional and active nature of procedures makes it possible to create interactive media which aid the user in determining what material is relevant, and facilitate the use of far more complex and fine-grained indexing mechanisms.

In summary, knowledge communication systems are distinguished from other computer systems in being a new *communication medium,* and from other communication media in that these systems are not only active, but also *interactive,* and have the potential to carry with them the communicative expertise of the designers in effectively conveying knowledge. Explanation research addresses the need for computational theories of how to select, reorganize, and present material to the users of such systems.

1.2 Preview of Issues

Of the many issues relevant to the development and implementation of a computational theory of explanation, several are of particular interest to myself, and are emphasized in this review. A brief statement of these issues will help the reader be alert for their manifestations before they are summarized in the final chapter.

Multiple Perspectives. Here the concern is with choice of conceptual framework within which to cast an explanation. For example, one can describe the operation of an electrical network in terms of constraint equations, or by describing causation between physical events, or simply describe its structure and leave it up to the hearer to infer its operation. The need for multiple perspectives demands more adequate representations for the kinds of knowledge one may wish to make available. The choice between perspectives should be made based on an understanding of which best fits the point being made, and on information about the user's background suggesting which perspective will be better understood. This leads to the next issue.

Epistemological Structure. The term "epistemological"[4] is used here to highlight an emphasis on theories of the nature of knowledge as the basis for understanding the structure of explanation. To put it concisely, my concern is with how an explainer exploits epistemological structure to meet explanatory and pedagogical goals. In particular, there is a need for an explicit computational theory of how:

- the internal structure of a domain's shared body of knowledge,
- the role of an individual's knowledge in understanding new concepts and situations, and
- the ways in which individuals are willing or able to transform this knowledge

[4]See footnote page 23 for a definition.

constrain the selection, organization, and presentation of the content of explanations. Many current theories of explanation simply describe explanatory structure without any theoretical justification. Principles are needed for both choice of perspective and for the finer grained selection and ordering of content within a perspective. The theory should indicate how to make these choices based on connections to the interlocutor's knowledge which enable him or her to understand and learn what has been expressed.

A Framework for Constraints on Explanation. One of the primary tasks of a theory of explanation is to define what a "good" explanation is. Such a definition may be expressed in terms of heuristic constraints which bear on the generation of an explanation. These constraints come from a variety of sources and impact on the explanation at different points in its generation. For example, I've already mentioned that an explanation must make contact with the interlocutor's knowledge, and that this impacts on both choice of the conceptual language used in expressing the explanation and on the sequencing of particular statements in this language. Other constraints derive from the goals of the interlocutor, and from human attentional and memory limitations, and impact on the logical and rhetorical structure of the explanation as well as its content. In the final chapter, I sketch a framework which delineates the sources of heuristics and constraints at various levels of analysis, and summarize how they impact on an idealized explanation generation process.

Coupling of Explanation to Other Aspects of Knowledge-based Systems. Much discussion at the AAAI-88 Workshop on Explanation was concerned with various manifestations of what I call the "coupling issue": to what extent should the design and operation of an explanation facility be coordinated with or mutually constrain:

- the design of the knowledge representation;
- the knowledge acquisition process and the contents of the knowledge base;
- the design of the problem solving architecture; and
- the operation and results of run-time problem solving.[5]

I point out these issues as they arise, and in the last chapter venture my own prescriptions.

Expressing a Theory of Explanation. Several methodological questions are concerned with what constitutes a useful medium within which to develop and express a theory of explanation. I discuss the relative merits of schemata, procedural specifications, and planning operators as vehicles for research in explanation. These devices provide the structure of the language used to articulate a theory of explanation. Another issue has to do with the choice of abstractions which provide the primitive terms organized by these devices. With the right abstractions, a theory is applicable beyond the domains studied in leading to its development, and implementations of the theory are more likely to be portable between architectures and representational paradigms. A third issue, which I barely touch on, is the extent to which one's computational architecture itself should constitute a theory of explanation by constraining what explanations can be planned, or whether architecture simply provides an interpreter which operationalizes the abstractions and computational entities used to express the theory. These issues are concerned with artificial intelligence methodology in general, and are not specific to explanation as a research topic.

[5]Cecile Paris and I prepared similar discussion questions for our panel presentation at that workshop.

1.3 Organization of the Paper

Each of the following paragraphs previews the chapter by the same name.

2. Beginnings and Influences. In the next chapter, I set the scene by introducing what I see to be several distinct sources of or influences on explanation research. Work on question answering, developed to evaluate cognitive models of understanding, and influences from linguistics are introduced here for later familiarity with their approaches, terminology, and problems. Early ICAI and ITS[6] research and attempts at explanation in automated reasoning systems are introduced in more detail to provide the groundwork for debate and improvements.

3. Epistemological Adequacy. In the third chapter, I focus on epistemological adequacy, i.e. making available the knowledge which will ultimately constitute the content of explanations. I start with two classic works which demonstrated the need for explicit representations of domain theory and problem solving knowledge in expert system explanation, in addition to surface-level rules mapping situations to actions or beliefs. I then look at ways in which research in ITS and other areas attempts to provide explanations taking a variety of "perspectives", "models", or "views" on the material being explained.

4. Explanatory Structure. Several levels of analysis have been used for describing the structure of explanations. In the fourth chapter, I examine two major contenders. I review some work focusing on how explanations take into account the interlocutor's knowledge in choosing explanatory content and its presentation, and/or facilitate the interlocutor's own active processes of modifying and extending his or her knowledge. These analyses are concerned with what I call epistemological structure. Analyses of the rhetorical structure of explanations and arguments uses abstractions which appear to mix logical, epistemological, and linguistic considerations with "purely rhetorical" devices for influencing the beliefs and attitudes of others. This work has provided a first pass at abstractions for describing the structure of explanations.

5. Generating Explanations. The fifth chapter is primarily concerned with alternate computational mechanisms for generating explanations, based on the analyses discussed in the previous chapter. I examine techniques for structuring explanations according to schemata derived from a corpus of natural explanatory text. Another approach is to use algorithms which exploit the structure of the represented knowledge itself. A discussion of planning in this chapter is motivated by the failure of schemata or algorithms to represent explanatory goals and assumptions in a manner allowing context-sensitive treatment of follow-up questions and recovery from failure of an explanation.

6. Issues for Explanation Research. The final chapter summarizes my observations with respect to the issues of section 1.2, and contains some suggestions concerning appropriate directions of research. In particular, I sketch the beginnings of a framework for integrating constraints on explanation at the variety of levels of analysis encountered in this review, suggest some appropriate research directions at the epistemological level, and comment on the expression of a theory of explanation.

[6]Intelligent Computer Assisted Instruction and Intelligent Tutoring Systems.

1.4 Limitations of the Review

The following topics are **not** covered, to the extent it is possible to avoid them while discussing the above. I acknowledge that these are relevant, but avoid them as too complex and involving too much additional literature.

- Text and graphics generation, screen layout, etc. Hence I am neglecting any constraints that available surface choices may place on content planning.
- Techniques for constructing user models; diagnosis of misconceptions, etc. A more complete treatment of explanation would include elicitation of feedback and diagnostic activities.
- Issues specific to the reasoning or problem solving task being explained. I assume for simplicity that a theory of explanation can be domain-independent. However, there may be explanation techniques which are unique within a domain or class of domains.
- Knowledge acquisition in support of explanation, and knowledge representation languages. Half of knowledge communication lies here. My treatment neglects any constraints the media, author, and authoring process place on explanations generated from the resulting representations.

Although research on tutoring strategies and the remediation of misconceptions is relevant, I have not made an effort to cover this literature except where it addresses the generation of explicit explanatory communications. Research in "explanation-based learning" (Mitchell, Keller, & Kedar-Cabelli, 1986; DeJong & Mooney, 1986) is not discussed because it is concerned with a different sense of "explanation" — i.e., accounting for an observation by proving how it follows from a general domain theory, which I consider to be a problem solving rather than communicative activity. Philosophical discussions of the nature of explanation (eg. Harrè, 1983; Quine & Ullian, 1970; and Taylor, 1970) and potentially relevant literature in educational psychology are also not discussed here. I attempt to limit the size of this review by remaining within the literature concerned with a *computational* theory of explanation.

Chapter 2

Beginnings and Influences

In this chapter, I introduce several distinct sources of and influences on explanation research, to provide background and present the issues to be discussed in their historical contexts.

2.1 ICAI

Early work on Intelligent Tutoring Systems (ITS), first called Intelligent Computer Assisted Instruction (ICAI), addressed a variety of problems with the goal of representing and teaching with computers. I am particularly interested in their concern with representing and selecting from multiple perspectives on the subject matter. Important contributions were also made in reasoning and representational mechanisms, user modeling and user sensitive interactions, and in natural language processing. Here I illustrate some of these contributions. Several overviews of the field are available, notably Wenger (1987).

2.1.1 SCHOLAR and WHY

Carbonell (1970) was the watershed between CAI and ICAI (which he called "information-structure-oriented CAI"). CAI relies on sequences of instructional interactions called frames. For each frame, the course author specifies exactly what the computer is to print onto the screen, how the student may respond, and what frame to go to next based on the response. All possible paths through the curriculum had to be represented explicitly, while the knowledge being taught and strategies for teaching it remained implicit. Carbonell's goal was to support more flexible mixed initiative dialogues, and his methodology was to define tutorial procedures which traverse a semantic network representation of knowledge (figure 2.1). SCHOLAR, a geography tutor, was the result.

For our purposes, one of the most significant contributions of SCHOLAR against the background of frame-oriented CAI was the separation of domain knowledge from control of the interaction. With this separation, tutorial strategies and lesson content may be acquired and changed independently of each other, and hence abstraction and study of a theory of explanation as an entity in its own right becomes possible. Also, the explicit representation of knowledge enables using it for a variety of purposes. For example, the semantic network was used as the source of knowledge to be taught,

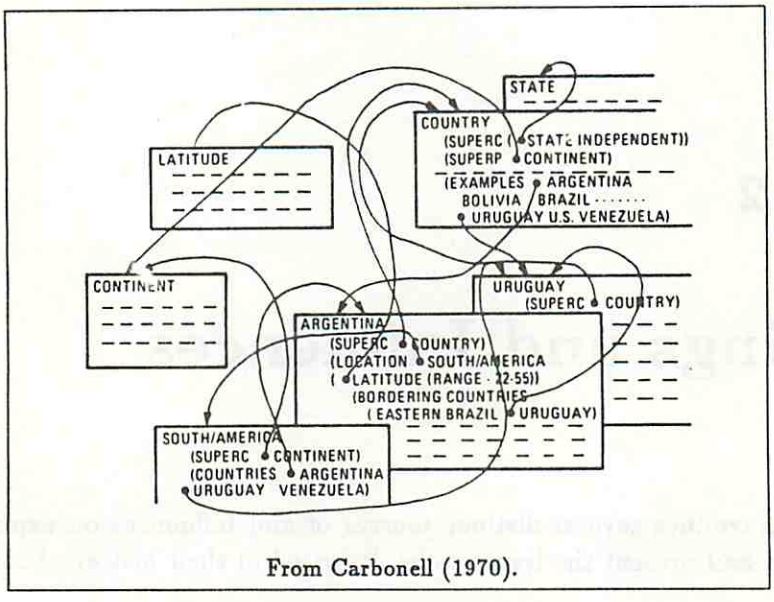


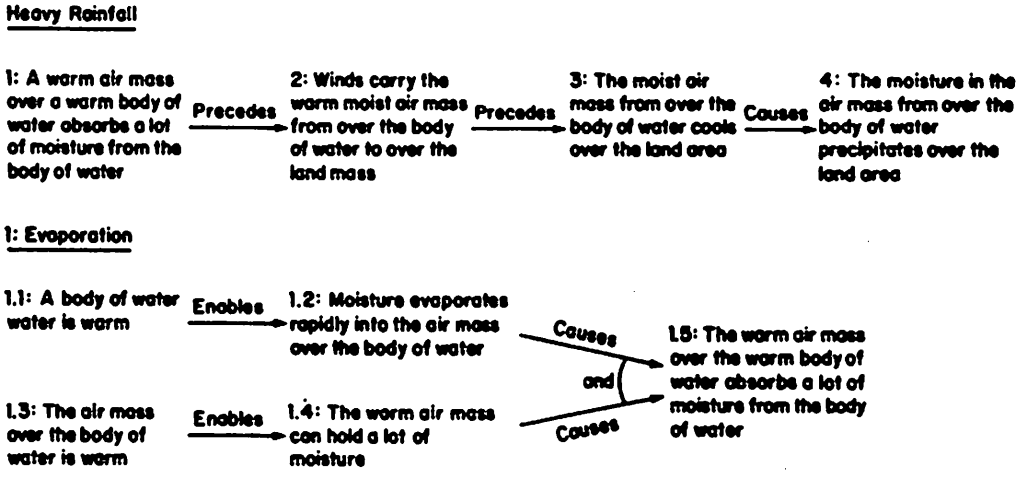
Figure 2.1: SCHOLAR's Semantic Network

as a standard against which to check student responses and, by modifying a copy of the network, as a representation of what the student knows. This latter use constitutes the first **overlay user model**, later named by Carr & Goldstein (1977) in the context of the WUSOR coaching tutor.

In comparing SCHOLAR with human tutorial dialogues, Carbonell and his colleagues found that SCHOLAR failed to make certain plausible spatial, temporal, and functional inferences, and to reason from contradictions or its own lack of knowledge. This was of concern because it prevented SCHOLAR from communicating knowledge which was implicitly available in its knowledge base through these inference, and furthermore because the researchers wanted SCHOLAR to lead students into making such inferences. As a result, Collins initiated his research on human plausible inference (Collins, Warnock, Aiello, & Miller, 1975; Collins 1978). Reasoning about the inferences an interlocutor is likely to make turns out to be important in planning the content of an explanation (section 4.1.3).

SCHOLAR's tutorial strategies were simplistic, relying on relevance weights and a semi-random topic selector with preference for recently mentioned content as it traversed the semantic network. The dialogues lacked coherence. Subsequent work (Collins, 1977) used heuristic rules to guide local topic selection. These were implemented in the WHY system (Stevens & Collins, 1977), resulting in dialogues with improved local coherence but lacking the global, goal-seeking strategies used by human tutors.

WHY's tutorial goal was to teach an understanding of processes leading to rainfall. In this domain, it is important to understand the sequencing and causality of events in terms of the processes generating these events. Semantic networks are not ideal for the representation of processes, so **hierarchical scripts** (figure 2.2a) representing sequences of events according to temporal and causal relations were used instead (Stevens & Collins, 1977). The hierarchy corresponded to different levels of detail in the causal model, and was traversed in a breadth-first manner to progressively refine the student's understanding. This is an early version of "model progression", discussed in section 4.1.4. However, scripts in turn are inadequate for representing the mechanisms behind rainfall. This led



(a) Scripts for heavy rainfall and evaporation.
From Stevens, Collins, & Goldin (1979).

EVAPORATION

Actors
Source: Large-body-of-water
Destination: Air-mass

Factors
Temperature(Source)
Temperature(Destination)
Proximity(Source, Destination)

Functional-relationship
Positive(Temperature(Source))
Positive(Temperature(Destination))
Positive(Proximity(Source, Destination))

Result
Increase(Humidity(Destination))

(b) Functional perspective on evaporation.
From Stevens, Collins, & Goldin (1979).

Figure 2.2: Perspectives on Rainfall

Stevens, Collins, & Goldin (1979) to discuss the need for a **functional perspective** (figure 2.2b) in which **actors**, related by functional relations, affect the outcome of processes according to the roles they play in those processes. They advocate using this perspective in combination with scripts and other perspectives, such as structural, physical principles, and metaphoric. The implications of this crucial shift from the search for a single good representation to an acknowledgment of the need for multiple perspectives will be pursued in section 3.2.

2.1.2 METEOROLOGY and SOPHIE

The METEOROLOGY system of Brown, Burton, & Zdybel (1973) was also a research effort in mixed-initiative CAI. It used a qualitative simulation model to represent processes, and was one of the first efforts in qualitative modeling. Causal knowledge about processes was represented using **augmented transition networks**[1] (ATNs). The ATN states were partial descriptions of meteorological states. LISP predicates tested meteorological conditions on the arcs, which were augmented with ways to compute answers and generate text. The student's question and a simple model of the student's current assumptions provide the context which initialized the state of the ATN. State transitions were then propagated, and enabled transitions recorded in an inference tree representing the propagation of state changes. This trace was then used to answer questions, the structure and coherence of the response being derived from the structure of this inference tree. Section 2.3 contains other examples of deriving an explanation from a reasoning trace, and discussion of the limitations of this approach.

Questions requiring justification of a single transition in the trace may be answered by applying the above technique recursively[2] to an ATN representing the associated process, resulting in a **reductionistic explanation**. METEOROLOGY also uses a semantic net in the style of SCHOLAR for providing factual information about the objects, concepts, and processes in the domain. This network is consulted first before invoking the process model. Brown, Burton, & Zdybel (1973) were "especially interested in the combined use of two divergent representations of knowledge". The use of hierarchically nested process models as well as their integration with propositional information make METEOROLOGY the first implemented system utilizing multiple perspectives. However, these workers had not yet moved to the epistemological level, where one considers the properties and inter-relations of the desired knowledge independently of their representations, so theory and implementation are not clearly separated. Stevens & Collins (1980) began to discuss models for meteorology at this level, but did not have a computational theory or implementation to back up their examples.

Brown & Burton subsequently went on to develop the SOPHIE series of simulation-based tutors. SOPHIE is known for its natural language interface, which was robust compared to systems of the time (Burton & Brown, 1979a). It is also another example of an early ITS system using multiple representations (Brown & Burton, 1975). It combined a quantitative simulation model, procedural specialists using this model, and a semantic network of facts, all of which were accessed by the natural language interface to answer questions posed by the student. However, there was no attempt at a theory of communicating multiple perspectives, the different representations being used for efficiency reasons only. Also, the simulation was a constraint-based "black box" which did not represent causality or handle assumptions. Its style of reasoning was not natural for humans, so it did not

[1] Nested finite state automata augmented with registers accessed by the tests on the arcs.

[2] This recursion quickly bottoms out in "explanatory notes", presumably canned text.

support the teaching of reasoning well. SOPHIE-II and SOPHIE-III switched to primary use of qualitative simulation, to meet the need for an **articulate expert**, whose operation is inspectable and appropriately structured for the purposes of explanation. This is an instance of the "coupling" issue introduced in section 1.2. Instead of rebuilding SOPHIE to reason like a human, Brown and Burton could have added a qualitative simulator and used it to generate a plausible causal account of the quantitative results. In fact, Brown & Burton (1975) discuss this possibility.

STEAMER, a large project influenced by WHY and SOPHIE, also suffered from reliance on a quantitative simulation. However, work on STEAMER led to interest in generating explanations from qualitative simulations, studies of multiple perspectives in explanation (Stevens & Steinberg, 1981), and other work to be reviewed later. Another product was DeKleer and Brown's (1983) work on mental models and qualitative physics, which resulted in important distinctions between different kinds of models, principles for construction of those models, and observations on the relations between assumptions and models and their use in explanation, subjects which I return to in sections 3.2.2 and 4.1.4.

2.1.3 Other Early ICAI Influences

WEST, the first computer coach,[3] used an "issues and examples" tutoring strategy where generic issues, which the student is believed to be having problems with, are illustrated by concrete examples to help the student integrate and retain the advice given (Burton & Brown, 1979b). Computational theories of when to use and how to choose examples is an important but neglected aspect of explanation (see section 4.1.2).

Contributions of the WUSOR project, a coaching tutor for a computer game, included the overlay approach to student modeling, and the **genetic graph** (Goldstein, 1979). The latter, which was influenced by Piaget's theory of genetic epistemology, attempts to represent in a graph the way knowledge evolves via transformations between rule-like units (see section 4.1.1).

Clancey's (1979) work on the GUIDON tutor led to advances in understanding the problem of epistemological adequacy. This work is reviewed in detail in section 3.1.1.

2.2 Natural Language Research

I mention several influences on explanation research from computational linguistics, and discuss influential work within an artificial intelligence approach to question answering. I make no attempt to cover these vast bodies of literature, emphasizing only a few crucial ways in which this research has addressed aspects of communication that can be treated independently of the syntax and lexicon of a particular natural language.

It is appropriate to start by mentioning **Grice's Maxims** (Grice, 1975), since these express constraints on acceptable behavior in conversation which are important considerations in the structure of explanation. Some examples of these maxims are:

Quality: Do not say what you believe to be false or for which you lack adequate evidence.

[3] "Computer coaches" are programs that observe a user's problem solving activity and occasionally offer advice.

Quantity: Make your contribution as informative as required for the current purposes of the exchange, and do not make your contribution more informative than is required.

Manner: Avoid obscurity of expression, avoid ambiguity, be brief, and be orderly.

Relation: Be relevant. Take into account the different kinds and foci of relevance and how these shift in the course of a conversation. Allow for the fact that subjects of conversation are legitimately changed.

There is a long ways to go from these informal maxims to a computational theory. However, they have served as a common goal for subsequent work in dialogue understanding and generation. In particular, research reviewed in the next section addresses the issue of relevance.

2.2.1 Focus of Attention

Linguistic studies of focus of attention in natural language dialogues have rightly influenced research on explanatory structure. This work has shed light on the way in which dialogue context determines the relevance of possible topics, brings some knowledge to mind while suppressing the rest, and constrains the way in which interlocutors move from one topic to the next. Understanding focus is crucial to the selection of relevant content and its linear organization into an explanation, as well as to disambiguating references in natural language utterances, the usual concern of the linguistic workers.

Grosz (1977) was influential in this area. Her research addressed what she called **global focus**, the center of attention throughout a set of discourse utterances sometimes called a "discourse segment". **Local focus** occurs at a finer granularity, shifting from sentence to sentence. Grosz pointed out that a major use of focus of attention is to order retrieval and deduction in dialog understanding so that relevant concepts and correct interpretations are more likely to be accessed first.

A distinction was made between explicit and implicit focus. **Explicit focus** consists of topics mentioned in the preceeding discourse. From this, the dialogue participants infer the **implicit focus**, which contains concepts that are relevant because they are closely connected to concepts in the explicit focus. To accomplish this, Grosz imposed visibility constraints on a semantic network with "vistas", which restricted retrieval to partitions of the network, ordered by relevance. Given a topic in explicit focus, the topic's superclass, the situation it occurs in (if any), and its subparts (if it is an object) were in implicit focus. Her heuristics for shift of focus were specific to task-oriented dialogues, and essentially followed the task/subtask hierarchy. The use of focus of attention for selecting relevant attributes came to be called "object perspective" in later work (section 3.2).

Sidner (1979) examined the use of local focus in the disambiguation of definite anaphora.[4] She tracked the current focus, the legal candidates for change in focus, and the past foci (using a stack). Her algorithms tell how to determine the referent of an anaphoric expression given constraints on when the speaker may continue the current focus, shift to a new topic (either explicitly or implicitly introduced), or return to a previous topic. These models were concerned with tracking focus of attention while understanding an ongoing dialogue, and said less about how constraints on focus of attention apply to the *generation* of an explanation. In section 5.1.1, I discuss McKeown's (1982) adaptation of Grosz and Sidner's work to generation in her TEXT system.

[4]Words such as "it" and "they" when used to point back to objects or concepts mentioned earlier in the dialogue.

Reichman (1984) has a more complex theory of how the content of a dialogue affects focus shifts. She attempts to formalize Grice's maxims as part of a generative theory of dialogue which accounts for our ability to change topics and track the structure of the conversation without needing explicit communications to coordinate agreement on that structure. In doing so, develops the concept of **context spaces**, categorizing them by the type of issue introduced or commentary, narrative, or evidential support offered in the context. Then, **conversational moves** are defined to be utterances (such as challenging a claim, explaining a claim, or shifting topic) which move between context spaces. The maxims show up as constraints on the applicability of these moves, and as expectations as to what subsequent moves are appropriate. Conversational moves appear to mix features of argument tactics (section 4.2.2), and speech acts, to which I turn.

2.2.2 Speech Act Theory

Analyses of language which equate semantics with truth value fail to capture the communicative intents behind utterances. Speech Act Theory (SAT) was developed by Austin (1962) to address this problem, and was further refined by Searle (1969) with influences from Grice. I follow Bach & Harnish's (1979) more recent version. The components of a speech act are:

Utterance Act: The physical act when a speaker (S) utters an expression from a language to a hearer (H) in a context.

Locutionary Act: The "literal", propositional content of the utterance: what S *says* to H.

Illocutionary Act: What S *does* through the utterance, where the act succeeds only if H recognizes the intention to produce the effect of the act.[5]

Perlocutionary Act: The effect S has on H, where H's recognition of the intention is incidental to the production of the effect.

The most theoretical effort has gone into the illocutionary acts. Bach & Harnish taxonomize these into four categories: **constatives** (asserting, confirming, concessives, disputing, predicting, suggesting ...); **directives** (requesting, prohibiting, permitting, ...); **commissives** (offers and promises); and **acknowledgments** (accepting, apologizing, congratulating, greeting, rejecting, ...). Each speech act is defined in terms of features such as what kind of propositional content is appropriate for the act, the attitude taken towards the locutionary content, situational preconditions for appropriateness, and expectations concerning what actions (speech or otherwise) are possible responses.

Speech acts (or their equivalent) have indisputable utility in mapping intentions to word choice in natural language generation. But what relevance does speech act theory have for planning explanations at a pre-linguistic level? The theory may indicate when particular communicative acts are appropriate relative to other acts, enabling computer participants in explanatory dialogue to act on explanatory intentions at appropriate points. Applications are seen in the literature on computer-supported cooperative work, for example Auramake, Lehtinen, & Lyytinen (1988) and Winograd (1988) use state transition graphs to encode legal sequences of speech acts in office "conversations" (figure 2.3). However, by viewing communicative behavior as purposeful action, speech act theory suggests a more fundamental shift in emphasis from the surface forms of natural languages to reasoning about the beliefs of the interlocutor and how communicative acts may change those beliefs. Under this emphasis, it seems more appropriate to take a planning approach to choosing speech

[5]See Bach & Harnish (1979) for a full discussion of the subtleties concerning recognition of intention.

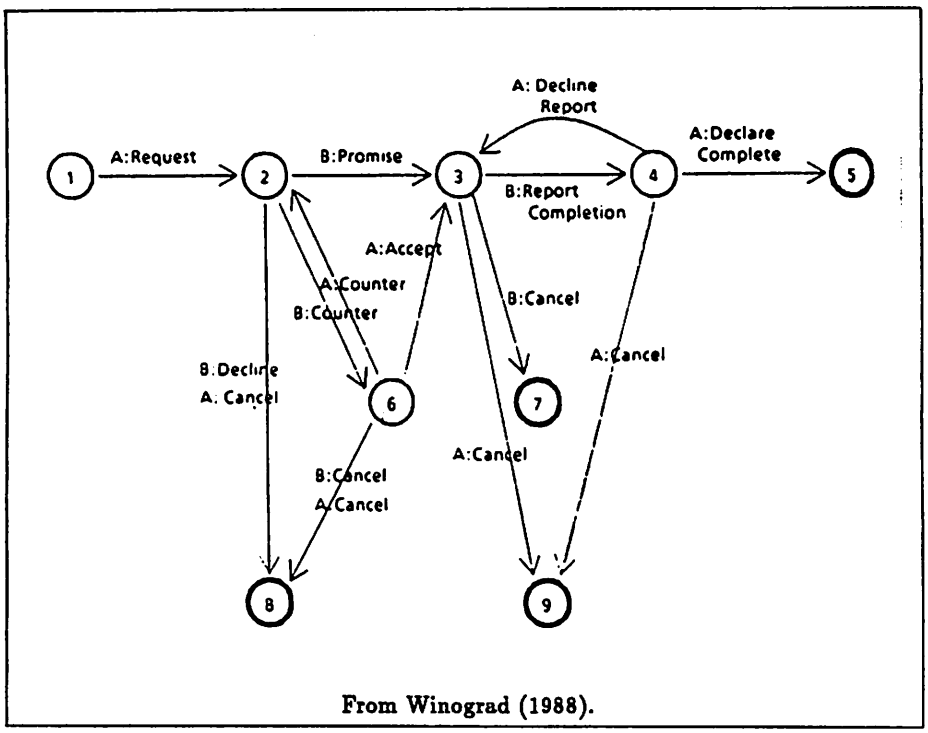


Figure 2.3: Graph for a "Conversation for Action"

acts according to how they meet goals of affecting the hearer's beliefs in certain ways, as Cohen & Perrault (1979) did in their seminal paper (see also section 5.2.3). (I will examine the relative merits of schematic vs. planning approaches to explanatory structure in Chapter 5.)

2.2.3 Question Answering and Taxonomies

In this section, I discuss primarily Lehnert's work on question answering, including her taxonomy of question types. However, I cannot fail to mention Woods' highly influential LUNAR system, a natural language question answering system for accessing a database of facts about lunar rocks collected by the Apollo program (Woods, Kaplan, & Nash-Webber, 1972). The techniques used have little direct import for understanding explanation, since most of the work was done in parsing restricted natural language questions into a database query language. However, the much-used ATN formalism was developed for natural language parsing by Woods in the context of this project.

Lehnert's research on cognitive models of understanding used question answering as a testbed (Lehnert 1977a,b, 1984). This is a tougher test of story understanding than paraphrasing or summarizing, since questions can probe for information not directly stated in but inferable from the story. Lehnert maintained that a taxonomy of question types will be most useful if it maps to procedures for answering questions. Since different procedures correspond to different ways to access memory, the question types will correspond to categories of memory requiring distinct memory access methods. Lehnert's theory of memory is based on Schank's (1975) conceptual dependency (CD), which has a strong relationship between semantic category and how memory is accessed. Hence her claim:

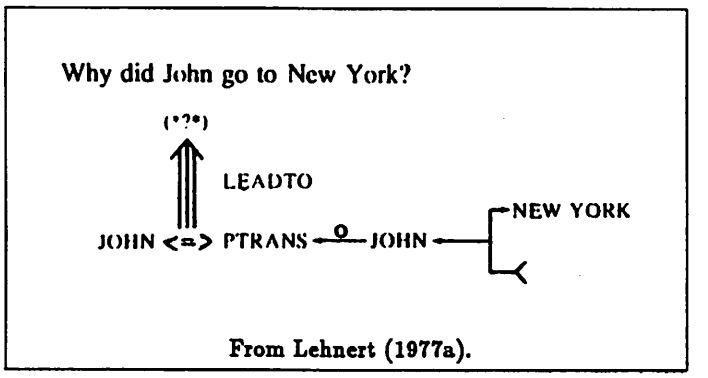


Figure 2.4: A WHY Question in CD

> "The problem of question answering is highly related to problems of memory representation and organization. It is therefore impossible to discuss question answering per se without reference to some model of memory." (Lehnert, 1977a)

At face value, this is an argument for strong coupling between a theory of explanation and the design of the representations being accessed. However, if the "model of memory" is an appropriately abstract one, some independence may be possible.

The basic process described by Lehnert (1977a,b) is one where a question represented in CD is classified into a question type, which is then used to choose the appropriate top-level procedure for invoking search of memory, the result being translated from CD into English. Much of the hard work is done in figuring out what CD structure expresses the question, taking into account the context and intent of the question. Once this is obtained, examination of the unfilled substructure of the CD question representation directly indicates what memory structures must be accessed to fill it. For example, a **Why** question is a causal chain (the causal relation may be one of motivation, goal orientation, or physical causation) where a consequent is specified but an antecedent unknown (figure 2.4). These are answered by providing the antecedent according to the appropriate causal relation. **How** questions are identified by an unfilled Instrumentality or Enablement role, which must be provided in the response. Most of the remaining work is embodied in the representation-specific procedures for obtaining a particular kind of information, guided by detail and elaboration options. While the classification process itself is not of great theoretical interest, its simplicity shows the power of CD as an interface language between the parser and the retrieval procedures, and more generally suggests that appropriate semantic abstractions are of value for communication at this interface.

Gilbert (1987, 1988) criticizes question taxonomies based on syntactic features of the questions themselves (he erroneously includes Lehnert), since the same information can be sought with different question formats, and different information with the same format. Instead, Gilbert prefers taxonomies of answer and knowledge types, where specification of the desired answer type leads directly to identification of the knowledge source needed to retrieve the answer. Neches, Swartout, & Moore (1985a,b) also taxonomize questions by the memory accesses required to answer them, but do so in terms of the architecture of an expert system. Hence, their taxonomy is useful at system design time for ensuring that the variety of knowledge types needed will be available, but it is not conceptual enough to support a theory of explanation at the desired level of abstraction. Tanner & Josephson (1988) provide a more principled analysis of the ways in which the user of a diagnostic expert system may question its operation, purely by virtue of the characteristics of the diagnostic task

itself. Chandrasekaran, Tanner, & Josephson (1989) generalize these observations to a taxonomy of generic task domains. This suggests that a useful taxonomy of questions (and perhaps a theory of explanation) *must* be specific to generic task domains.

2.3 Explanation from Problem Solving Traces

Not long after ICAI researchers were translating semantic networks and scripts into tutorial explanations, other workers began investigations into translating proofs or justification structures into English text. Around the same time, the builders of rule-based expert systems developed simple question answering facilities for describing and justifying their system's reasoning.

These efforts had much in common. Explanations were typically in response to a small number of categorical questions, each being mapped to a method of accessing the problem solver's data structures. The problem solver's trace provided the initial content and structure of the explanation, modified only by transformations for comprehensibility and editing of undesirable portions. The comprehensibility transformations were concerned with the logical structure of the explanations, and with managing the complexity of chains of reasoning. Sensitivity to the user's prior knowledge and goals was implemented by filtering certain types of information (eg. rules required solely for implementation reasons) from the problem solver's trace; by using a depth cutoff in the goal tree to suppress detail; and/or by removing assertions present in a database modeling what the user knows.

I provide some details of their operation to give a flavor of the terms in which theories of this type were expressed: the principles for organizing the logical structure of explanations are given only implicitly, in terms of operations on the problem solver's data structures. Inadequacies which were subsequently discovered in these pioneering attempts provide motivation for discussion in the next three chapters.

2.3.1 Chester's EXPOUND

Chester's EXPOUND (1976) generated multi-paragraph English proofs from a representation of their formal counterparts. The task of translating formal proofs into written proofs was a "clean" domain in which to study of discourse structure in extended text, since the formal proof provided a semantically well grounded representation of the "deep structure" of the written proof.

A Natural Deduction[6] system was used to generate the formal proofs. EXPOUND translates the proof into text in four passes. The first pass creates a graph representing inferential relations. The graph represents a partial ordering, and is constructed using graph operations encoding constraints such as: "State reasons before conclusions" and "Subproofs should be nested". The second pass creates a paragraph-level outline by grouping vertices in the previous graph together into sets of vertices corresponding to paragraphs. The third pass completes a linear order of the lines in a paragraph, and inserts introductory paragraphs to clarify relations between the other nodes. On the fourth pass, text is generated using templates for paragraph structure and translating logical connectives, and lexical information associated with the predicates.

[6]Unlike axiomatic approaches which use only one rule of inference, natural deduction systems allow conditional proofs, and use a variety of inference rules which distinguish different ways in which statements are deduced from their predecessors.

The resulting text has a uniform level of detail, and tends to have excess detail, making it hard to follow. An improvement would be to structure the proof hierarchically, allowing a summary of structure and optional expansion to detail. EXPOUND cannot do this because it understands the proof primarily at the level of detail at which it is formalized, with minimal use of more global patterns. For example, it could not say "Similarly for case 2" as mathematicians often do. Chester was interested in the structure of extended text, not interactive communication. Because of this, he makes inadequate use of the potentially multidimensional and interactive nature of the computer as a communication medium.

Chester is to be credited for developing straightforward and efficient structuring manipulations on dependency graphs, and working out the details of the most obvious tricks (not described here) one can exploit in going from logic to English. The graphical manipulations do have the advantage of being content independent and hence generally applicable. However, as a theory it is at the wrong level of analysis. Some of the heuristics for structuring the text were obscure and seemed unmotivated, because they were expressed in terms of logical and graph properties rather than in terms of the underlying principles of communication which make these manipulations reasonable. For example, a (paraphrased) heuristic for paragraph construction is:

> Add lines at node x to those at y if y is an immediate successor to x; x has y as its only immediate successor; x has less than 5 lines in it so far; x has no conditional derivation lines (which end paragraphs); no immediate predecessors to y should be added first to ensure subproofs are unbroken sequences; and x has the fewest number of immediate predecessors.

I believe that appropriate abstractions will afford content independence where it is merited while also clearly expressing the theory of communication.

2.3.2 Weiner's BLAH

Weiner (1980) describes BLAH, a system which provided an explanatory interface to the AMORD reasoner. Interactions with BLAH were user initiated, using the commands `SHOW` (whether believed), `EXPLAIN` (why believed), and `CHOICE` (between alternatives). Explanations were generated from support trees provided by AMORD's **truth maintenance system,**[7] which contain several different kinds of justification links (`statement/reason`, `if/then` for hypotheticals, and, or, `general/specific`, `examples`, and `alternatives`). For example, to `EXPLAIN`, BLAH would:

1. Create a tree representing the statement and its support.
2. Remove from the tree all assertions assumed to be known by the user.
3. Determine level of detail; possibly break the tree into subtrees.
4. Generate text from resulting tree and subtrees.

Most of the paper is about editing and restructuring this support tree so that the text generated from it will be easier to follow.

[7] TMS: an algorithm which records support relations between assertions and uses them to propagate changes in belief and detect contradictions.

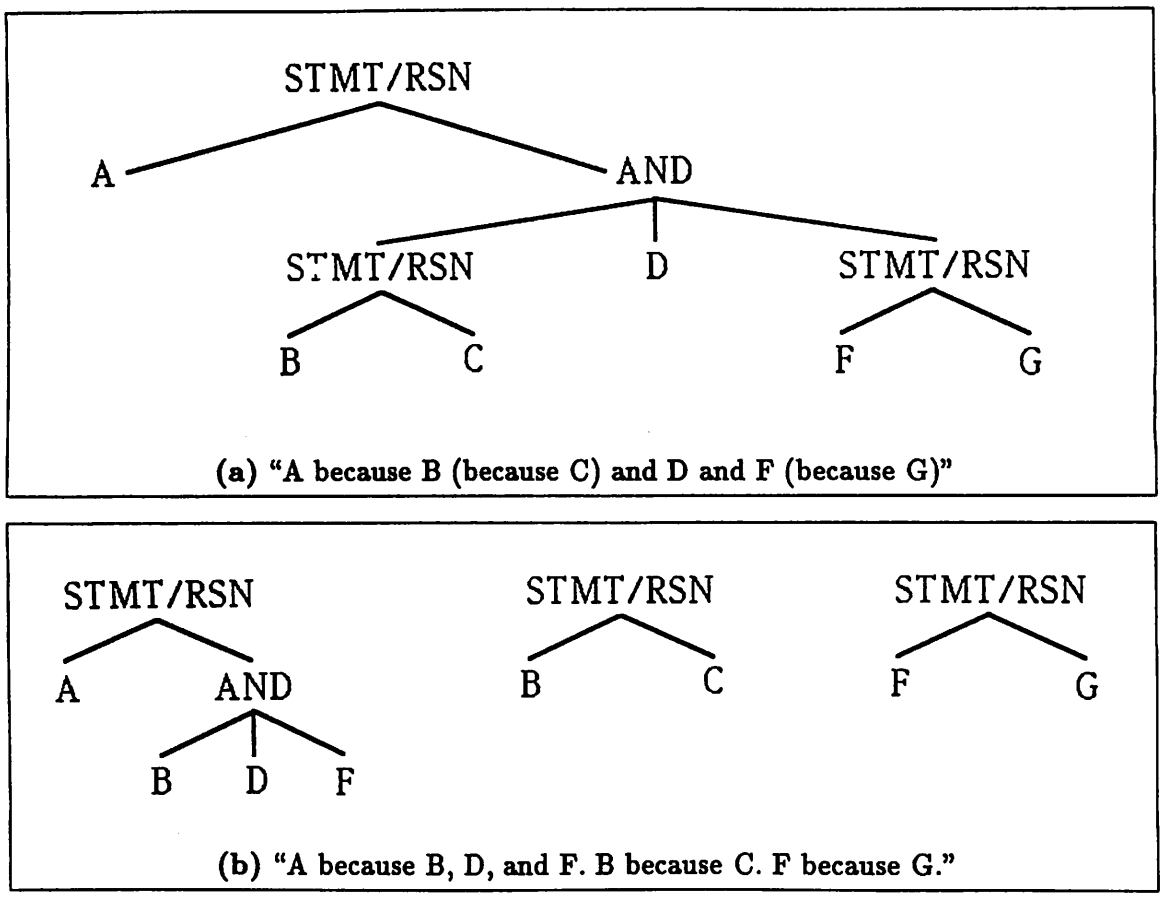


Figure 2.5: A Comprehensibility Manipulation in BLAH

Pruning Based on the User's View. Unlike EXPOUND, BLAH attempted to be sensitive to the difference between the user's and the system's beliefs. A "view", or partition of the set of assertions, "contains the information (assertions and rules) that its owner (i.e. system or user) knows". The user's view can be reasoned with just like the system's. Anything provable in the user's view is assumed known. (Unless such proof uses a restricted proof system modeling inferences humans draw reflexively, there is a potential problem with logical omniscience.) Pruning is a right-to-left, bottom-up elimination of any justification tree nodes which BLAH can show the user knows. In general, a node is not deleted unless all nodes below it are, so BLAH can give a new reason for a known assertion. If the explanation gave only one reason when there were several, the user could get the false impression that it was the only reason. Hence there are heuristics which retain information even though less would be logically sufficient.

Example of a Comprehensibility Manipulation. The surface level text cues the reader of when focus shifts occur. For example, conjunctions are cued using "First of all, ... Second of all, ...". However, even well-cued focus shifts become unmanageable if much information is presented in the subtrees before returning to subsequent conjuncts or disjuncts. To deal with this, BLAH breaks up the tree below each n-ary node which has at least three sons, at least two of which are non-terminals. (If there is only one non-terminal, it can be moved to the right, where it will not distance other

terminals from each other.) Each non-terminal son becomes the root of a new subtree, to be given after the main tree. This has the effect of reducing the complexity of a deeply nested explanation, as shown in figure 2.5.

Limitations. Weiner criticized BLAH for its lack of a natural language interface; lack of knowledge of the pragmatics of communication; limitations in its model of reasoning; too strong assumptions about the user not forgetting what has been said; poor criteria for choosing examples (he did not say what criteria he used); and the inadequacy of depth in the justification tree as a measure of detail.

One of my concerns is his dependence on particular representations (rather than epistemological and rhetorical level descriptions) to express his theory. However, the representation is abstract enough to be applicable to many systems. There is a tension (in my mind at least) between the clarity of his theory with respect to what has to be done to his data structures, and its lack of clarity as a principled theory of explanation. His transformations and editing activities are the "compiled" versions of constraints and heuristics having to do with human attentional and memory limitations, and what constitutes an informative response. BLAH knows nothing about these constraints and heuristics, and hence is the result of rather than the articulation of a theory of explanation. Also, this approach to explanation is strongly tied to the justification trees, and is weak on providing other kinds of knowledge and examples. Though some work continues in this style (e.g. Eriksson & Johansson, 1985), it is not likely to be productive towards a theory of explanation.

2.3.3 The WHY and HOW of MYCIN

MYCIN, the prototypical rule-based expert system, is described in Davis, Buchanan, and Shortliffe (1977). This paper reveals that explanation was one of the fundamental motivations behind the development of the symbolic rule-based paradigm:

> "One of our original design criteria, then, was to give the system the ability to provide explanations of its behavior and knowledge. It soon became evident that an approach relying on some form of symbolic reasoning (rather than, for example, statistics) would make this feasible. This was one of the primary reasons behind the choice of the production rule representation ..."

It is ironic that today MYCIN is often referenced as the classic example of how *not* to do explanation in expert systems. Certainly the choice of a symbolic representation of knowledge was right, but as we shall see in Section 3.1, the crux is what you represent, and MYCIN simply failed to be epistemologically adequate for explanation.

Explanation in MYCIN, like many rule-based systems to follow, paraphrased rules in response to two question types: WHY and HOW. If the user typed "why" in response to a question, MYCIN explained why it was asking the question by looking up the goal tree, stating the goal the current rule is relevant to, then paraphrasing the rule (indicating which sibling subgoals had already been satisfied) to show how the information asked for would enable the rule to address the goal. Repeating "why" resulted in this explanation procedure being re-applied to the next higher goal-rule-subgoal link. When asked how a conclusion was reached, MYCIN simply listed the rules which were used. Asked how a rule was used, it described the evidence which satisfied the rule, and then stated the conclusion of the rule. Repeated invocation of "how" stepped down the goal tree. These facilities

could be applied prospectively ("how will you ...") as well as retroactively, and the explainer could simulate MYCIN's control algorithm to answer "why didn't you ..." questions. Davis's TEIRESIAS (1979) applied these facilities to explaining and debugging MYCIN-like systems during knowledge acquisition.

In the context of techniques available at the time, MYCIN was an advance in capabilities of explaining the knowledge behind a system's reasoning. It produced acceptable explanations for a user familiar with both the domain and the rule-based architecture. Success was attributed in part to the use of symbolic reasoning, a generally accepted conclusion within the explanation and ITS research communities (recall SOPHIE's articulate expert, section 2.1.2). However, Davis, Buchanan, & Shortliffe (1977) also claim that:

- IF/THEN rules are a natural way of expressing domain knowledge, so display of such rules should be helpful.
- Backward chaining and depth-first search of AND/OR goal trees are intuitive, and hence comprehensible.
- Since the rules came from experts, they:
  - embody accepted, understandable patterns of human reasoning, and
  - are in the right size "chunks" to be comprehensible.

These conclusions may be challenged on several fronts. As discussed further in section 3.1.1, MYCIN's rule base itself proved inadequate as a basis for explanation in GUIDON, a tutorial program (Clancey, 1979; Clancey & Lestinger, 1981; Clancey, 1986a). Rules were contaminated by control clauses attempting to get the interpreter to do the right thing, and different types of clauses were not distinguished as to their purpose. The rules were designed to reach the right conclusions, and did not include knowledge unnecessary for doing so but essential to teaching a student to act as primary diagnostician. More fundamentally with respect to the above claims, rules are not the most appropriate representation for much of the knowledge required for explanation, and some knowledge is better communicated in ways other than stating rules. The multiple representations of section 2.1 illustrated this close relationship between a good representation for a piece of knowledge, and a good way to present it. Chandrasekaran, Tanner, & Josephson (1989) argue that both problem solving architectures and the constructs used to explain them will be closer to the user's conceptualization of the task if designed in terms of the properties of **generic task domains**, such as "hierarchical classification", "design by plan selection and refinement", "state abstraction", and "knowledge-directed information processing", rather than in terms of implementation level objects such as rules and frames. Even if one grants that backward chaining on rules is intuitive, controversy continues concerning whether experts reason this way (Patel & Groen 1986), and even whether an emphasis on the problem solving process facilitates learning (Sweller, 1988). Finally, what constitutes appropriate content and "chunk" size varies with the user, the dialogue context, and the nature of the knowledge being communicated, so no single-granularity representation will be adequate, even if the grain size and content is chosen by experts.

2.3.4 Evaluation

Explanations generated from simple transformation of proofs or program traces are easy to implement and virtually guaranteed to be consistent with the program's reasoning (a significant advantage over use of "canned text", i.e. separately recorded text strings). However, the appropriateness of the

explanation depends greatly on how the rules or programs are written. The domain model on which the program's reasoning is based may be inappropriate for a given user, or may be designed to optimize the computations at the expense of explainability. As illustrated in research about to be discussed (section 3.1), the program trace supports explaining what the system did, but not why. Other knowledge and reasoning was used by the programmer in writing the program, which should be accessible for generating explanations justifying the program's behavior. Program traces don't support other knowledge communication activities such as defining terms, conveying concepts and principles, etc. All these considerations point to the need for representations of knowledge besides that strictly needed by the problem solver (consistent with the ICAI research in section 2.1). The necessity of coherence manipulations such as in BLAH suggests that a program trace is not always the best guide to explanatory structure. Chapter 4 will discuss alternate ways to structure an explanation. Finally, research which focuses on techniques to get explanations out of program traces tends to result in a compendium of architecture and representation dependent techniques without abstraction to a theory of explanation.

Chapter 3

Epistemological Adequacy

The research reviewed in this chapter is concerned with identifying and representing the knowledge drawn on for the content of explanations. First I review two seminal works addressing the inadequacy of early expert systems for supporting explanation. These researchers characterized epistemological[1] research as that concerned with "*the knowledge that is required to solve a problem* and *the aspects of problem solving behavior that need to be explained*" (Hasling, Clancey, and Rennels, 1984). However, full understanding of a domain includes richly interconnected concepts, examples, and results (Rissland, 1978) as well as knowledge specific to problem solving. Continuing the trend towards increased richness of knowledge first illustrated by ICAI research (section 2.1), I examine current work in providing and using multiple perspectives or models. These enable an explainer to use a conceptual framework which best fits the material to be communicated and the background of the recipient. The issue of how to choose and order perspectives in an explanation will be taken up in the next chapter.

3.1 Epistemological Adequacy in Expert Systems

Working at about the same time in different types of expert systems and applications, William Clancey and William Swartout came to similar conclusions: expert systems can't explain themselves if certain knowledge in addition to that needed for run-time performance is not represented. Both discuss the need for abstraction of control knowledge and representation of the structural and causal relations which that control knowledge uses to index performance rules. Both redesigned expert system architectures to do this, though in different ways.

[1] "Epistemology" is "a theory of the nature of knowledge" (*American Heritage Dictionary*). The term originally identified "the branch of philosophy which investigates the origin, structure, methods, and validity of knowledge" (*Dictionary of Philosophy*). The latter reference also outlines the major issues of philosophical epistemology, viz. is knowledge possible; if so, what is its origin and what are its limits; how do you get it; what kinds of knowledge are there; what is the relation between the knower, the content, and the object of knowledge; and the nature of truth. In this review, I use this term with a somewhat more psychological emphasis than the philosophers, being less concerned with the relation of knowledge to reality, and more with the psychological reality of how knowledge functions for the individual seeking to understand something.

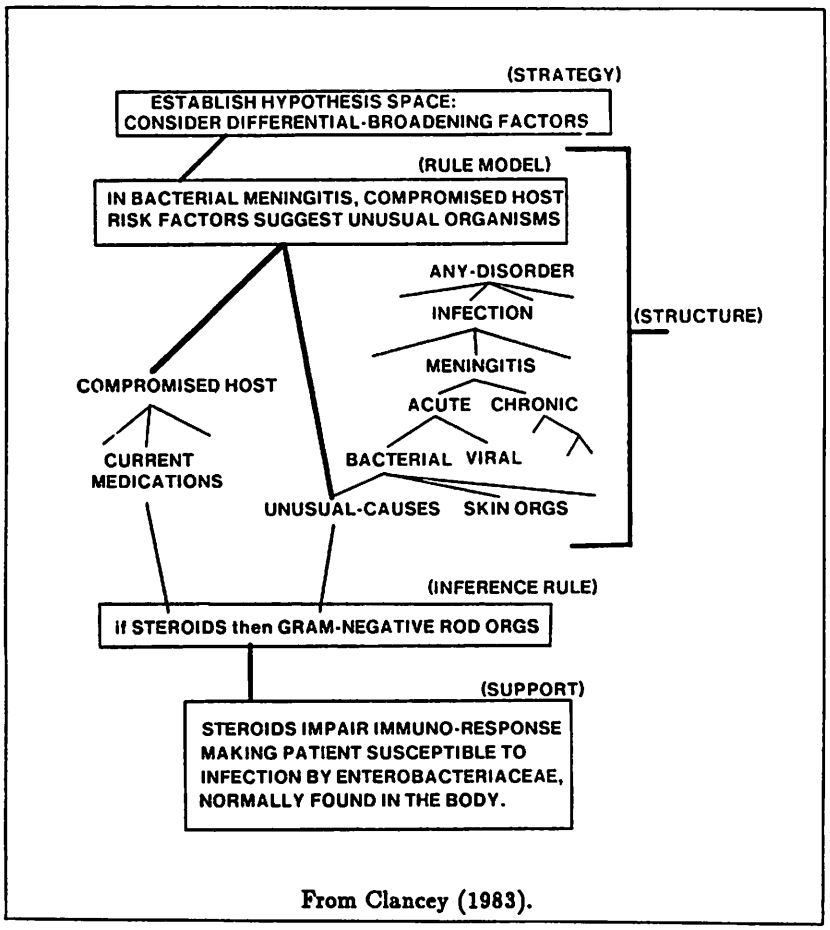


Figure 3.1: Indexing with Structural Knowledge

3.1.1 Teaching Diagnostic Strategy: MYCIN to GUIDON-II

Clancey's attempt to build the GUIDON tutoring system as an interactive front end on top of MYCIN exposed various problems with the MYCIN paradigm (Clancey & Lestinger, 1981; Clancey, 1983; Clancey, 1986a). These were mainly problems of missing knowledge. There were no justifications for rules in terms of how the problem features involved are related. The rationale for the order in which subgoals of a rule or rules for a goal are pursued was not represented anywhere, a serious deficiency, as this rationale constitutes the problem solving strategies which must be articulated in explanation and tutoring.

Abstract Strategies and Structural Knowledge. NEOMYCIN and GUIDON-II were developed to address these problems (Clancey & Lestinger, 1981; Clancey, 1983; Hasling, Clancey, & Rennels, 1984). Clancey observed that a **strategy** is a domain independent plan by which goals and hypotheses are ordered in problem solving. Strategies are related to domain knowledge by **structural knowledge**, which includes subsumption and etiological relations and knowledge about diseases considered as processes. One can state the strategy in terms of abstract categories in a subsumption hierarchy, and use the structure to index to domain concepts (and hence rules) satis-

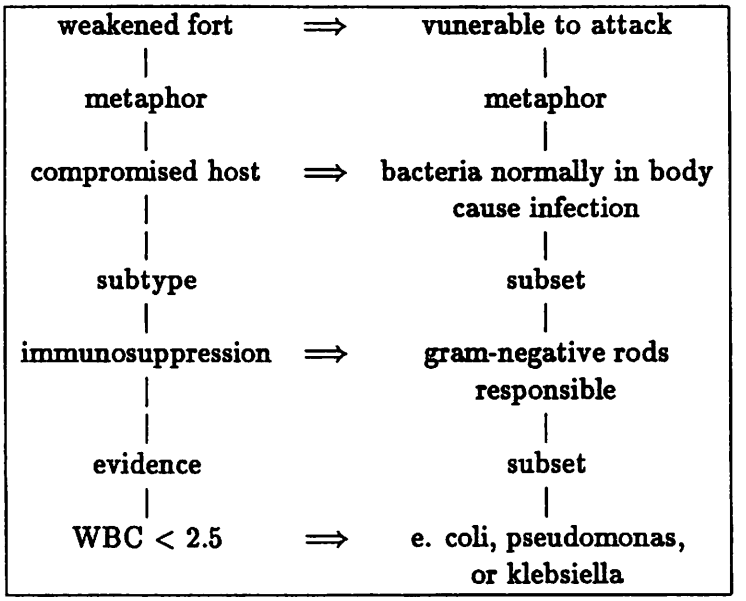


Figure 3.2: Abstraction to Familiar Principles

fying the strategy. "In effect, the layer of 'structural knowledge' allows us to separate out *what* the heuristic is from *how* it will be used" (p. 239 of Clancey, 1983). Figure 3.1 illustrates this indexing. As also shown, rules are justified by supporting them with knowledge about causal processes.

The explicit representation of this extra knowledge supports multiple perspectives in tutorial explanations. A disease can be classified, or the instances of a given disease category enumerated. One can describe how a disease evolves as a process, and give a causal account of this process. Clancey (1986a) puts special emphasis on teaching the way diagnostic strategy decomposes an abstract hierarchy of diagnostic tasks, and on the domain relations by which the terminals of this task hierarchy index knowledge sources. This is in contrast to the discredited idea in GUIDON that production rules are the the right units of knowledge to communicate.

Like the earlier tutoring systems of section 2.1, the need for a variety of knowledge was dealt with by introducing multiple representations to NEOMYCIN. However, Clancey's analysis (especially Clancey, 1983) is an epistemological one, where the types and properties of knowledge needed for explanation are analyzed, rather than methods for representing that knowledge. His work is significant as a move towards the right level of theoretical discussion, as much as for its contributions towards understanding physician's explanations and building diagnostic rule-based expert systems capable of communicating them.

Explanation as Abstraction to Familiar Principles. A satisfying explanation must make contact with already known concepts. According to Clancey (1983), this means that to explain a rule one should generalize it until it is transformed into a familiar, common sense pattern of reasoning at an "almost metaphorical level." Figure 3.2 provides an example (modified from one in his paper). The specific rule at the bottom of the figure is explained by relating it to a more intuitive principle by going up 'evidence' and 'subtype' links on the left hand side and down 'subset' links on the right hand side. Thus, one justifies the rule "white blood count less than 2.5 is evidence that one of e. coli,

pseudomonas, or klebsiella is responsible" by noting that a person who has been weakened is more likely to be infected by bacteria found in his body (implicitly relying on the metaphor shown), and showing how the rule is an instance of this general principle. This is in contrast to a reductionistic style of explanation which uses nested causal models (as in METEOROLOGY, page 10) to expand a rule into detailed physical theory.

I suspect that the tutoring application biased Clancey towards relying mostly on such commonsense principles. The intuitive patterns of reasoning he abstracted towards are likely to make contact with any student's knowledge. However, though the metaphorical understanding may be helpful at first, certainly for some users and dialogue contexts a reductionistic explanation would be more appropriate. When does one explain by reducing to finer-grained theory, and when does one show how it fits into a familiar abstract pattern? Or when does one not abstract or reduce at all, relying instead on analogies or case-based explanations? These questions have not been adequately addressed, but see section 4.1.4 for some comments on the choice of an appropriate perspective in explanation.

Methodology for Epistemological Adequacy. Clancey (1983) provides a methodology for designing epistemologically adequate representations, and applies this methodology to an analysis of several major expert systems. The process is one of identifying the knowledge sources (recognition or construction operations) used; determining the knowledge structures by which they are indexed or organized and the theory which justifies them; and then ensuring that these structures and justifications are represented. In Clancey (1986a), he describes this as an iterative process: write statements in some language; organize it and classify patterns observed; try to explain the patterns in terms of primitive relations; and if necessary change the epistemology by defining a new language allowing these primitives to be stated explicitly and used to generate the original patterns. Tutorial explanation compelled the development of this methodology for epistemological adequacy. In the next section, I look at an alternate methodology which serves a similar function.

3.1.2 Justification via Automatic Programming: XPLAIN

Swartout (1983) was concerned with justifying the actions of the Digitalis Therapy Advisor:

> "By justifications, we mean explanations that tell why an expert system's actions are reasonable in terms of principles of the domain – the reasoning behind the system. The knowledge required to provide justifications is not represented in typical expert systems because the program can perform correctly without it." – p. 287.

In particular, the missing knowledge includes a model of causality in the domain, and the reasoning principles by which one uses this model to make inferences. Swartout employed an automatic programmer which generated the performance system using the following knowledge structures. The **domain model** contains descriptions of facts of the domain, such as causal relations and classification hierarchies. This is Clancey's "structural knowledge". **Domain principles** are the methods and heuristics of the domain, expressed as abstract procedural schema, to be filled in with facts from the domain model to yield specific procedures. Though more complex in structure, these are similar to Clancey's "abstract strategies". They have these parts:

Goal: This tells the automatic programmer what the principle can accomplish.

Constraints: Conditions which must be satisfied if the principle is to be used.

Domain Rationale: A pattern matched against the domain model which provides additional information necessary for achieving the goal (presumably by variable binding). The pattern itself also functions as a template for generating text which justifies procedures generated from the principle.

Prototype Method: An abstract method which tells how to accomplish the goal. Each match of the domain rationale to the domain model tells the automatic programmer how to instantiate the prototype method (using the bindings).

Viewpoints: These are labels attached to steps in prototype methods to indicate who these steps should be explained to when an explanation is being generated from an instance of the prototype. (There were two viewpoints: "computer" and "medical".) This allows the separation of "computer artifacts" from the domain justifications. Swartout claims that the domain principles are an appropriate place to mark these viewpoints because that is where implementation and domain knowledge is brought together, and one doesn't have to annotate every instantiated method.

The automatic programmer uses the domain model and principles to generate a **refinement structure**, or tree of goals, each being a refinement of the one above it, and the lowest level goals being primitive actions.

Generating Explanations. XPLAIN can describe both prototype and instantiated methods, explain how the system drew conclusions, and say why a question is being asked. For example (from Swartout, 1983), suppose XPLAIN asks *"Is the patient showing signs of paroxysmal atrial tachycardia with block?"* and the user asks *"WHY?"*. The justification routine goes up the control stack, starting explanation at the first procedure which is not in an excluded viewpoint and which has not been called by an excluded viewpoint. (The problem of deciding which viewpoints to present to a given user was not addressed.) If, for example, the viewpoints specify a medical audience, XPLAIN skips implementation level procedures found in the control stack. The explanation starts by paraphrasing the chosen higher level procedure, to say what the system is doing in general: *"The system is assessing the highly specific findings of digitalis toxicity."* If the higher level procedure was generated by matching the domain rationale of its corresponding domain principle to the domain model, this match is used to generate an english description of the rationale: *"Increased digitalis may cause paroxysmal atrial tachycardia with block which is a highly specific finding of digitalis toxicity."* XPLAIN finishes off the explanation by paraphrasing the current procedure: *"Thus, if the system determines that cardiomyopathy exists, it reduces the dose of digitalis due to cardiomyopathy."* The explanation assumes the user knows that paroxysmal atrial tachycardia is a disorder of the myocardium, and uses the more general term "cardiomyopathy".

If the user asks for further justification, XPLAIN moves up the control stack to make the higher level procedure the current level, in a manner reminiscent of MYCIN. However, this can get abstract in the "wrong way".[2] Some explanations must use abstraction selectively to express the relationship between general and specific knowledge. For example, to describe a domain principle that produced some code, he describes the domain rationale with the refinement variables replaced by their matches, but with the domain pattern variables described as themselves.

[2]One wonders if Swartout's wrong was Clancey's right, namely, abstraction toward very general principles.

Evaluation. In response to the above "why" question, a MYCIN-like system would have said something like "The system is trying to determine whether it should reduce the dose of digitalis. Paroxysmal atrial tachycardia is evidence (0.75) for reducing the dose of digitalis [Rule037]." Note that this is similar in content to XPLAIN's last sentence in the above example. XPLAIN adds to this by justifying the action with a causal model, and by reference to the more general domain principles which generated it. Thus the improvement is epistemologically similar to that of NEOMYCIN over MYCIN.

A user who didn't know (or infer) the relationship between cardiomyopathy and tachycardia would have been mystified by the above explanation. Swartout acknowledges that the explanation generators in XPLAIN could be extended to focus on the domain model itself, and take on a tutorial role to improve the user's understanding of this knowledge apart from its use in problem solving. He suggests that such a capability would require user models, tutorial strategies, better English generators, and an understanding of the relation between the system's explanatory capabilities and the user's question.

Swartout also discusses telling "white lies" (deliberate oversimplifications) for pedagogical purposes. However, he has a poor idea of where these come from. He suggests retaining old versions of principles and methods when they are modified to give better results. These earlier fragments would then be the source of white lies. This ties explanatory simplifications to program development in a way which will likely result in totally inappropriate simplifications. In sections 3.2.2 and 4.1.4, more principled sources of deliberate simplifications are described.

The Role of Automatic Programming. Swartout originally thought that automatic programming would be too hard, and the development trace would have to be created by hand along with the code.

> "However, as the research progressed, it became clear that if we had sufficiently powerful representations available so that it could be said that in some sense explanations were being produced from an understanding of the program, then actually writing the program in the first place would not be much more difficult. I suspect this is true in general. It seems that the primary difficulty in both explanation and automatic programming is a knowledge representation problem, and that the kinds of knowledge to be represented in both cases are similar so that a solution to one makes the other much easier." - 319.

Swartout goes further to report that "improvements in the quality of the explanations generated resulted more from the use of an automatic programmer than from increases in the sophistication of the [English] generation system." The attempt to find the principles that generate the program forced them to consider their methods more closely, and discover flaws in the original Digitalis Therapy Advisor (Swartout, 1977). The automatic programmer requires representation of appropriately abstracted principles and causal knowledge, and it leaves a trace of the justification relation between the abstract knowledge and the performance procedures. While Swartout credits the automatic programmer, the essence of the improvement is one of epistemological adequacy. Clancey was led to very similar representations and conclusions via attempts to use an expert system for tutoring. Hence, while explanation requires *some* knowledge acquisition methodology to achieve epistemological adequacy, automatic programming is not the only source of discipline towards this end. The automatic programming methodology does have significant advantages for the maintenance of expert systems. Dhar & Pople (1987) leverage this advantage further, by synthesizing the qualitative domain model itself in a manner sensitive to the "task environment".

Explainable Expert Systems. This work continues as the "explainable expert systems" (EES) project at ISI (Neches, Swartout, & Moore, 1985a,b). Certain types of system development knowledge which were not represented in XPLAIN (and hence had to be applied as developer intervention into the automatic programming) are being incorporated in EES. This includes **tradeoffs** between domain principles, **preferences** by which choices are made between tradeoffs, and **integration knowledge** used to resolve conflicts. Unlike XPLAIN, if no direct implementation is found for a goal, the EES program writer can reformulate the goal in various ways discussed in Neches et al. (1985a,b). This will enable the explainer to make more subtle distinctions concerning the justifications for the program's behavior. On the generation end, XPLAIN's schematic approach is being replaced by explanation planning techniques, reviewed in section 5.2.3, to enable more flexible explanations and utilize dialogue context when answering follow-up questions or recovering from failure to communicate.

3.2 Multiple Perspectives

In section 2.1, I reviewed early work in tutorial applications concluding that multiple representations are needed for different kinds of knowledge to be taught, and that there is sometimes a discrepancy between the model of a domain used for a computer-based problem solver and that most appropriate for explanation. These points were also illustrated in the expert system explanation work just reviewed. It seems clear that sophisticated knowledge communication systems will require coordination of multiple perspectives on the subject matter. In this section, I discuss various AI research efforts, not normally grouped together, which I consider to be concerned with multiple perspectives. My goal is to define and explore this notion, and indicate why perspectives are important for explanation. The relevance of some of this material will become clearer in section 4.1.4, where I discuss the choice and ordering of perspectives for pedagogical purposes.

Informally, **perspectives**[3] are different ways of conceptualizing an entity or process. Examples are forthcoming. Abstractly, I suggest that a perspective is to be characterized by the ontological or epistemological[4] distinctions it makes. This characterization may be formalized in terms of the elements and interpretation of a language for expressing the distinctions; i.e. in terms of the set of primitives (such as terms, predicates, functions, and relations) used, along with the intended interpretations of these primitives. Distinct perspectives differ on at least some of their primitives and on how the range of the interpretation function objectifies the domain of discourse into objects, properties, and relationships. A perspective is to be distinguished from a set of beliefs in a manner

[3]A note about terminology. I avoid using "model" as synonymous with "perspective" for several reasons. In the literature I review, a "model" is almost always a runnable simulation, something visualisable in the mind's eye, or an analogy. "Model" is also used to mean "interpretation which satisfies a set of logical formulas" by logicians. Though "view", "viewpoint", or "viewtype" is used by some writers in the same sense as I intend "perspective", a "viewpoint" is at times restricted to mean an *individual's* way of looking at things. In Suthers (1988a,b), I have used "view" to refer to what you get when you apply a perspective to a particular topic. Barbara Gross (1977) calls this a "vista" . In the literature review, I always use the author's terminology when first introducing the work.

[4]If one takes the subjectivist philosophical position that one cannot speak of what exists independently of an agent's knowledge of it, then "epistemological" is the more accurate, meaningful term, and indeed subsumes ontology (for what it's worth). Though some perspectives I discuss are "ontological" because the conceptual framework is applied to understanding of things believed to be "out there", i.e. physical entities and events, I prefer to characterize the concept of perspectives as an epistemological one, to focus on distinctions in the conceptual structure of our *knowledge* essential for planning explanations.

analogous to the syntactic distinction between a language and a set of expressions in that language. While a formal characterization of perspective must rely on syntactic distinctions, the notion of perspective itself is an epistemological one.

Some definitions equate choice of perspective with choice of a subset of the available predicates saying something about a given entity. For example, McCoy's (1985, 1986) **object perspectives** are sets of attributes weighted with salience values dictating which attributes are highlighted when responding to misconceptions about objects. Stronger notions of multiple perspectives include **ontology shifts**, i.e. perspectives may differ on what entities there are in the first place as well as (and consequently) on what may be predicated of them. Superb examples come from the ICAI literature in the domains of meteorology (Stevens & Collins, 1977; Stevens, Collins, & Goldin, 1979; Stevens & Collins, 1980; section 2.1.1), steam propulsion plants (Stevens & Steinberg, 1981), and electrical networks (Frederiksen & White, 1988; White & Frederiksen, 1987). This literature appears to converge on a distinction between **structural perspectives**, where the objects are the *physical parts* of a device or system, and some form of *adjacency* and *constituency* are the primary relations of interest; **causal perspectives**, where *states* of a system are *causally* related; and **functional perspectives**, describing *processes* in terms of *simultaneous constraint equations* between *state parameters*, or alternately, *functional relations* between *actors*. I will not claim that there is a clear line between simply partitioning the set of predicates and radical shifts in ontology. Many of the knowledge communication issues which arise may well be the same.

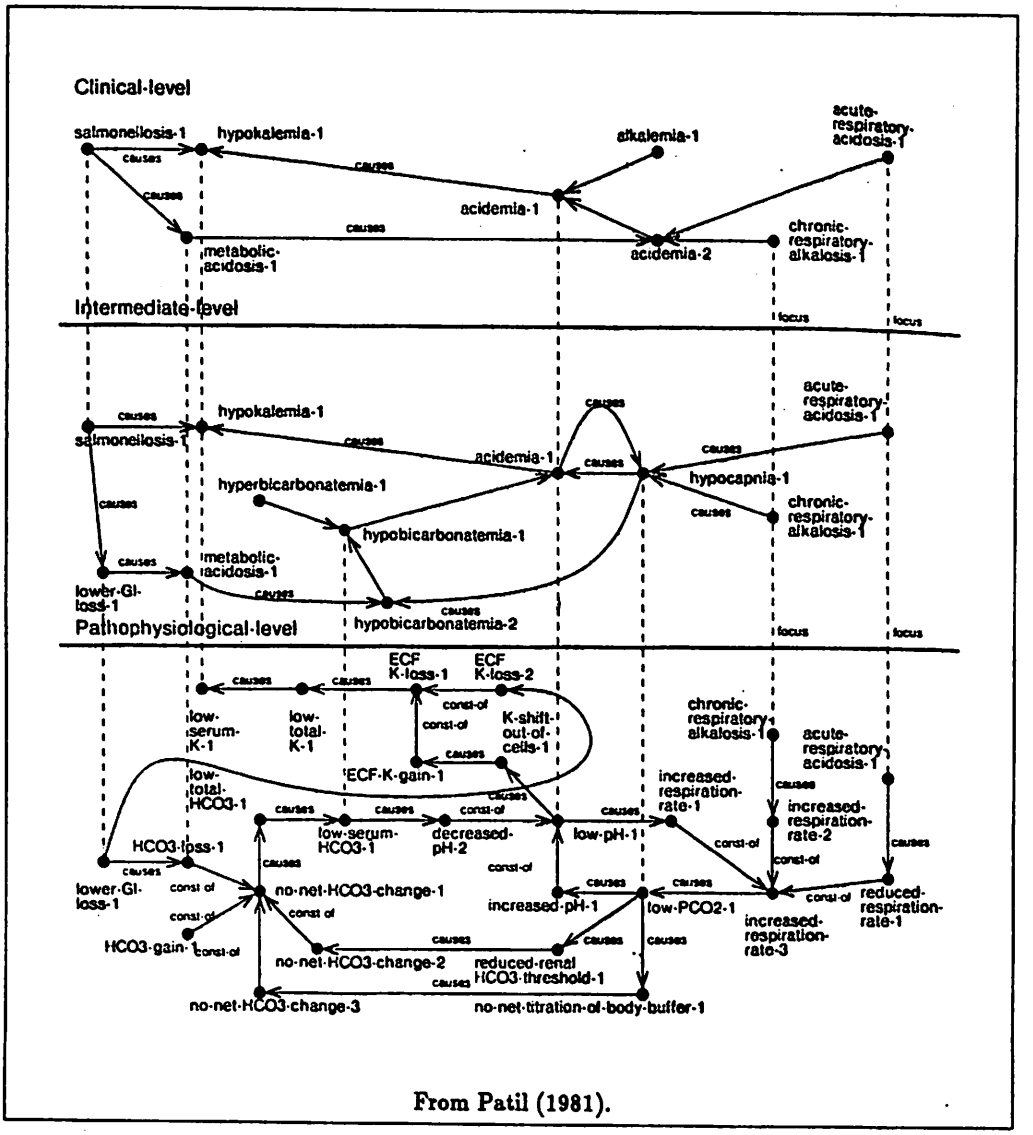
I will start with some examples illustrating various types of perspectives and relations between them. Subsequent sections will provide further examples, emphasizing mechanisms for representing and generating perspectives, and how perspectives couple explanation to problem solving. I summarize the relevance of perspectives for explanation in section 3.2.4, but leave further examination of their role in explanatory structure to section 4.1.4.

3.2.1 Examples of Perspectives and their Inter-relations

The literature reviewed in this section was chosen to augment the preceeding abstract discussion with more concrete examples of perspectives. It illustrates the range of available perspectives, some relationships between them, and a variety of uses to which perspectives may be applied.

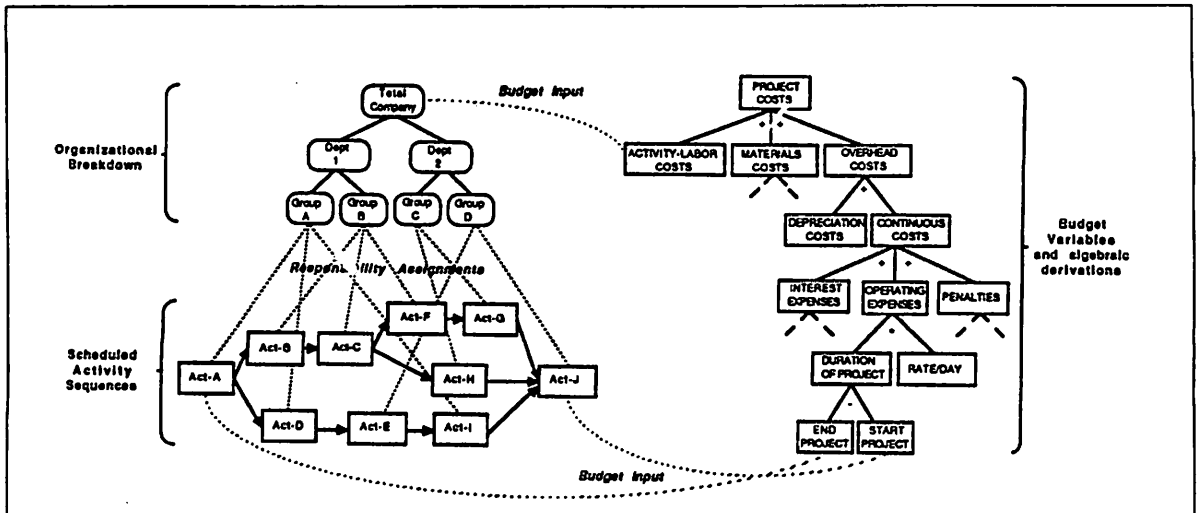
Hierarchical Perspectives for Problem Solving. Patil's ABEL (1981), a medical diagnostic system with a causal model layered into clinical, intermediate, and pathophysiological levels, provides an example of hierarchically organized perspectives (figure 3.3). The hierarchy expresses the direction of inference from clinical observations to pathophysiological diagnoses. It also helps reduce the complexity of problem solving: clinical reasoning occurs at a coarser granularity and guides search in the more detailed levels. An explainer in such a domain needs to decide at what level of detail the explanation will be given. Unfortunately this was not addressed by ABEL's explanation facilities (which were based on Swartout's early work).

Heterarchical Perspectives for Indexing Quantitative Information. Roth, Mattis, & Mesnard, (1988) provide a good example of how different "models" can be integrated to interpret and explain quantitative data. They start with a mathematical model of cost change in a large industrial project. The mathematical model says what variables changed, but doesn't group them in a



From Patil (1981).

Figure 3.3: Hierarchical Perspectives for Problem Solving



From Roth, Mattis, & Mesnard (1988).

Figure 3.4: Organizing Quantitative Information

useful manner. Roth, Mattis, & Mesnard add an organizational hierarchy, a temporal graph of how scheduled activities depend on each other, and an algebraic graph describing how costs break down (figure 3.4). Together, these models are used to explain the results of the quantitative model by identifying which qualitative categories or events seem responsible for the cost changes. Incidentally, much of the paper addresses the issue of coordinating graphics and text in explanation, and is worth reading for this. There may be a close relationship between choice of perspective and of the medium in which it is presented.

Dimensions of Perspectives on a Physical Device. Stevens & Steinberg (1981) taxonomized explanations used in Navy manuals for steam propulsion plants. The taxonomy reads more like a taxonomy of perspectives one could take on a physical device:

Componential descriptions[5] identify the parts of an object.

Topological descriptions say how the parts of an object are connected.

Geometric descriptions are topological ones with quantitative information added.

Behavioral explanations describe the input/output mappings of a component.

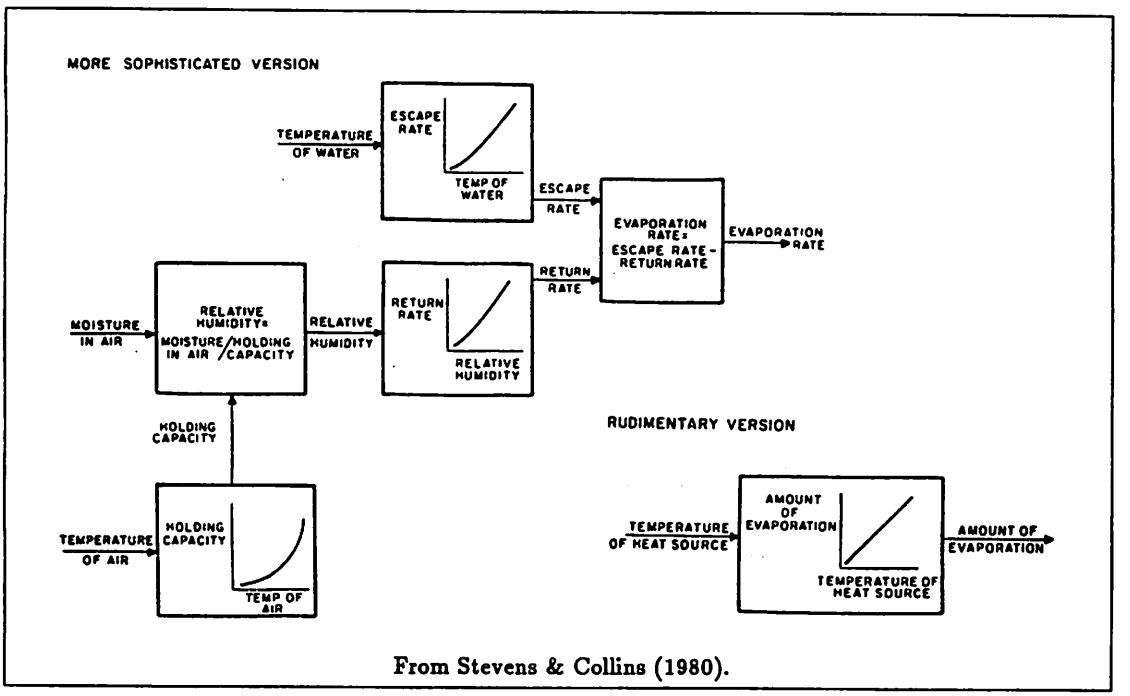
Synchronous explanations describe causation in terms of simultaneous constraints.

Physical-Causal accounts view the system as a causal chain of events. It is a discrete version of Synchronous, possibly involving fictional time units.

Information Flow explanations provide a cybernetic feedback view of the system.

Stuff/State/Attribute explanations are in terms of how substances get moved around and changed in the system.

[5]For hearers who can infer function from structure, a description counts as a process explanation: see the discussion of qualitative reasoning in section 3.2.2.



From Stevens & Collins (1980).

Figure 3.5: Simple and Refined Functional Models of Evaporation

These perspectives can be compared along certain dimensions. In the progression from Componential to Geometric, first qualitative and then quantitative information is added. All three of these emphasize structure whereas the others describe mechanisms. Stevens & Collins analyze their taxonomy in terms of such dimensions, namely **qualitative to quantitative**; **structure vs. mechanism**; **individuated vs. aggregated** (properties of single objects vs. groups or "stuff"), and **external vs. internal properties** ("black boxes" vs. "glass boxes"). These dimensions were used to identify what perspectives might be missing, and may have further utility as abstractions for indexing and selecting the perspective appropriate for a given explanatory task.

Relations Between Perspectives in Meteorology. Following up on Stevens, Collins, & Goldin (1979, see section 2.1.1), Stevens & Collins (1980) provide some well connected examples of "multiple conceptual models" in meteorology, and suggest different ways in which these models may be related to each other.

Simulation models can be used to answer questions about processes. For example, a simple analogical simulation model of evaporation describes it as if water and air molecules were billiard balls in motion. A more sophisticated simulation model includes a force of molecular attraction between the billiard balls. This increases explanatory power, for example, unlike the previous model, it explains the surface tension of water. This is an example of the **refinement relation** between two perspectives: the perspectives are essentially the same, but further information has been added to one. The **functional perspective** on evaporation describes functional relations between the input and output variables of the evaporation process. A simple version says that rate of evaporation is linearly related to the temperature of the water. Refinements take into account other factors such as air temperature and the amount of moisture in the air (figure 3.5).

Stevens & Collins (1980) emphasize the importance of being able to map between simulation and functional models. The simulation model's mechanisms **justify** the relations described by the functional model; while the functional relations provide a **summary** of these mechanisms, providing two more relations between perspectives. Finally, they describe models of the role of evaporation within the water cycle, which in turn take part in global climate patterns. This illustrates the **component relation** between models: evaporation is a component process in the water cycle model, and water cycles depend on global air movements.

Analogical Perspectives. Gentner & Gentner (1983) and Clement (1988) focus on the use of analogical models for understanding physical entities and processes. In analogies, knowledge and associated terminology about already understood entities provides the conceptual framework within which a new entity is described; hence analogies may be seen as a special kind of perspective. For example, one could liken an electrical circuit to a plumbing system, where electricity is water, batteries are pumps, and resistors are constrictions in the pipes. Analogies are useful because knowledge about the familiar source analogue may be used to make inferences about the target; e.g. fluid flow through a tighter pipe constriction can be made up for by a stronger pump, so a battery with more voltage balances a stronger resistor with respect to current flow ($I = V/R$). Hence, an explanation can potentially convey a lot of information with a carefully chosen analogy. The danger is that the interlocutor may not know the limits of the analogy, and make inappropriate inferences based on unintended transfer of properties from source to target analogue. For example, noting that capacitors store electricity and water is incompressible, one might infer incorrectly that once a capacitor is filled, one cannot store more electricity in it by using a stronger battery. This suggests that explanations relying on analogies should provide guidelines in their use. However, if one has to detail all permissible inferences, the analogy looses its economy of expression, if not its role in making the unfamiliar familiar. An explainer might at least guard against incautious use of the analogy by emphasizing its nature as a hypothetical way of thinking about the target, rather than as an ontological statement.

Further examples of multiple perspectives in human reasoning about physical systems may be found in Clement (1978); Collins (1985); Collins & Gentner (1983); and Williams, Hollan, & Stevens (1983).

3.2.2 Representing and Generating Perspectives

In this section, I discuss some mechanisms for representing or generating perspectives, and comment on issues relevant to their utility for explanation. I digress momentarily into knowledge representation issues because the need for multiple perspectives is one of the major challenges explanation places on knowledge representation technology, and because these attempts to represent perspectives also illustrate different ideas of what perspectives are.

Inheritance Methods. An early approach to object perspective, used in KRL (Bobrow & Winograd, 1977), was implemented by changing the class an object inherits from. A variant on this was Wilensky's (1984) "views" in the KODIAK representation language. These allow one to define new concepts by treating one concept as if it were "dominated by" (inherits attributes from) another concept. For example, the concept of body may be defined as a `person` viewed as a `physical-object`,

meaning one considers primarily those attributes of the person which are physical-object attributes. Also, a view includes a context which may contain arbitrary assertions not normally true, but which are considered to be true when this particular viewpoint is taken.

McCoy (1985) criticizes the inheritance method on two grounds. To ensure that the perspective is right, one must not only get the object to inherit from the right superclass but ensure that this superclass inherits from the right super-superclass, and so on all the way up to the top of the class hierarchy. This criticism applies only to systems in which inheritance is transitive and attributes cannot be filtered according to their origin. McCoy also claims inheritance fails to represent perspective as something taken on all the objects of a domain (as opposed to a single object). In response, it is unclear whether one *needs* to be able to apply any given perspective to all objects in the domain. There does not seem to be an a-priori reason why one could not construct a class representing a perspective which is a superclass of all objects to which the perspective applies. Then, addressing the first criticism, it is possible to select and use just those attributes whose existence is inherited directly from the perspective class. The appropriateness of inheritance for explanatory perspectives depends less on problems of mechanism, which can be dealt with, and more on whether the perspectives one needs can be defined in terms of fixed sets of attributes indexed by class names. The body example in KODIAK suggests that such a mechanism works for perspectives defined relative to specific concepts such as physical object. More abstract perspectives (e.g. "functional") can only be defined in terms of sets of *attribute classes*, since the names of particular functional attributes may change across different types of objects. The inheritance approach also needs a mechanism for changes in granularity (discussed below).

Attribute Weights. An alternate approach is to use weighted associations between attributes. McCoy (1985, 1986) is concerned with highlighting the appropriate attributes of objects when responding to misconceptions about them. Each of her perspectives is a frame containing a set of attributes with associated salience values, dictating which attributes are highlighted and which are suppressed. Only one such perspective is active at a given time, and all objects accessed by the system are viewed through the currently active perspective. The perspective is chosen by focus of attention mechanisms[6] (section 2.2.1). A perspective becomes active if a "few" of its highly weighted attributes have been mentioned in the current focus. Then subsequent explanations will highlight attributes, mentioned or not, which are heavily weighted in the chosen perspective. Thus, a perspective is represented as weighted associations between collections of attributes. McCoy's paper only provides a mechanism for using perspectives in similarity comparisons: the salience values are converted into weights on the attributes as they are entered into a similarity metric function.

McCoy's mechanism may be adequate for representing a wide variety of salience patterns at a fine level of detail, and it avoids the problems for which she criticized KRL and KODIAK. She also provides a mechanism for choosing an appropriate perspective. However, her contribution towards a theory of the use of perspectives in explanation is limited. Her mechanism cannot describe the epistemological contribution of perspectives, and her theory attempts no generalization across domains. This is consistent with her standpoint that "knowledge of useful perspectives ... is part of the domain expertise" (McCoy, 1985). Other researchers believe one can have a domain-independent understanding of perspectives and their utility in problem solving and explanation, and have attempted generalizations on this basis (Stevens & Collins, 1980; Stevens & Steinberg, 1981; Souther, Acker, Lester, & Porter, 1989; Suthers, 1988a,b; White & Frederiksen, 1987).

[6]McCoy was influenced by Grosz's (1977) early comments on object perspective.

The Logical Approach to Granularity. Hobbs (1985), concerned with computational tractability in logical theories, provides a formal approach to abstracting approximate theories of a coarser granularity from a complete theory. He assumes one has a **global theory**, called T_0, which contains all there is to be known at the finest level of granularity, and describes how to extract smaller, more tractable theories from it. In the process of **simplification**, one selects the subset of predicates relevant to the current task, collapses objects into equivalence classes of objects which are indistinguishable under these predicates, and redefines the domain and range of the predicates to be over these equivalence classes. Hobbs also discusses **idealization**, which provides a way to eliminate inherited indistinguishability of equivalence classes when mapping to simpler theories. He mentions the **articulation problem** of how all these theories relate to each other, but unhelpfully concludes that "it is largely a matter of spelling out the particular cases in the knowledge base".

A problem lurks here: T_0 is too much to ask for. It is convenient to be able to define local theories with respect to a complete, global one, but what if the latter does not exist? Humans learn the simplified theories first, and only over time elaborate on them and weave them together into something that begins to approach a global theory. A way to characterize the simplified perspectives and how they "articulate" with each other without resorting to T_0 is a prerequisite to expressing a theory of how such perspectives are selected and coordinated in explanation and learning. One possible solution arises from the use of assumptions.

Assumptions in Qualitative Reasoning. Some background on qualitative reasoning is relevant. DeKleer & Brown (1980, 1983) distinguish the following models and processes in their theory of qualitative reasoning about physical devices. The **device topology** represents the structure of the device. **Envisioning** is the process of inferring function from this structure together with descriptions of component behavior. Envisioning would normally result in a **causal model** describing this functioning in terms of how changes in device states propagate to other such changes. However, ambiguities often arise in the envisioning process, due to the qualitative nature of the reasoning and to contextual assumptions in the component models. So instead, envisioning results in an **intrinsic mechanism**, which represents all the possible causal models, indexed under the assumptions required to produce them. Unless assumptions are made, the only way to disambiguate the intrinsic mechanism into a single causal model is to make use of functional evidence from the device's actual behavior, in a process called **projection**. Finally, a given causal model may be used to answer specific questions about behavior by **simulation**, i.e. running it from a given initial state.

There is a tradeoff between problem solving robustness and efficiency. A causal model is **robust** if its predictions are faithful to the behavior of the actual system, even in unusual situations. Since these situations cannot all be foreseen and tested, one must rely on a set of principles for robustness while constructing the device topology and component behavior rules. The most central is the **no function in structure** principle, which requires that "the rules for specifying the behavior of any constituent part of the overall device can in no way refer, even implicitly, to how the overall device functions". This turns out to be difficult to achieve, and results in highly complex models, encumbering problem solving. Experts deal with this by violating the principle, deliberately introducing **assumptions** to envision simplified simulation models. Thus, experts reason and explain with models which on the surface appear to be similar to those of novices, but which have assumptions attached as caveats. Understanding the role of assumptions is an essential part of understanding problem solving in the domain, so assumptions should be explicit and explainable. However, the importance of assumptions is also epistemological. They provide clues to how to simplify a body of knowledge so that it may

be communicated in comprehensible units. DeKleer & Brown draw some conclusions concerning the implication of this for explanation, discussed in section 4.1.4.

3.2.3 Coupling of Perspectives and Reasoning

This section illustrates and discusses how perspective couple explanation to decisions made by an automated reasoner.

Perspectives in Diagnostic Reasoning. I have already mentioned how Patil's ABEL uses hierarchically organized perspectives to represent the direction of diagnostic reasoning and control its complexity. Perspectives are also useful as alternate problem representations, where each may prove to be most perspicuous for a given class of problems. For example, Davis' (1984) approach to fault diagnosis of digital circuits coordinates a purely logical perspective describing the intended behavior of the circuit with a geometric representation of distance between wires, to determine the location of possible shorts. He discusses how the usefulness of different representations of the circuit arises from their underlying notions of **adjacency**, i.e. electrical, thermal, and electromagnetic pathways of causation between components. The choice of explanatory perspectives in such domains will largely follow from which perspective was most useful in problem solving.

Perspectives as Alternate Reasoning Methods. McKeown, Wish, & Matthews (1985) give the following examples of "points of view" for explanations in a course advising domain:

Requirements: "Take this course because it's required for your goal course."

State Model: "Take it because it is usually taken the semester you are in."

Semester Scheduling: "Take it now because it isn't offered next semester."

Personal Interests: "Take it because it is relevant to your interests."

Perspective is applied by selecting the appropriate attributes to focus on when describing or reasoning about a given object. Each of these viewpoints corresponds to a distinct hierarchy in the knowledge-base. The appropriate hierarchy is chosen by applying goal inference techniques to a set of user utterances to identify the user's goal in a predefined lattice of goals, each goal having a pointer to the object perspective hierarchy to use. Then a production system is run, using only assertions from the chosen hierarchy, and a response is generated from the result of the reasoning. Thus, a perspective is a distinct way of reasoning.

By assuming hardwired indices from goals to perspectives, McKeown, Wish, & Matthews have not shed theoretical light on this selection. They give no provision for mixing perspectives, or for including information which is informative yet not necessary for the production system to reach conclusions. Coupling the style of reasoning with explanatory goals does produce more appropriate problem solving traces for generating the explanation. It works when the granularity at which reasoning occurs is the same as that at which one would want to switch viewpoints. However, one cannot extract different viewpoints after the reasoning has occurred, which may be a disadvantage in some applications. Explanation in Bienkowski's EXTEMPER (1986) can switch between **rehashing** previous problem solving and **rehearsal**, which directs problem solving towards explanatory goals, effectively deferring a decision on the coupling between explanation and problem solving from design time to run time.

Choosing and Coordinating Assumptions in Qualitative Simulation. Hobbs did not indicate how to choose the set of relevant predicates in simplification. Falkenhainer & Forbus (1988) address this in a system which explains the behavior of a steam propulsion plant. They are also motivated by computational tractability, since their unsimplified causal models could not run to completion on the resources of a LISP machine. Envisionment of an appropriate causal model is guided by explicit assumptions which throw away irrelevant parts of the system being modeled, combine other parts into abstract components, and select the relevant grain size for answering the question at hand. The result is a simplified causal model which is still adequate to serve the desired purpose. Qualitative simulation of the model is then controlled by assumptions which filter out irrelevant behaviors.

"Views" provide the mechanism for choosing the appropriate assumptions. When the conditions of a view are met (e.g. the reasoner is directed to consider the thermal properties of physical objects), then the view specifies which predicates and relations over physical objects to allow in the model (the thermal properties). Since several of these views may be composed to generate a model, there is a problem of ensuring that local decisions made in the views generate a coherent model. Falkenhainer & Forbus deal with this problem by representing and enforcing dependencies between the views. The domain model is divided into subsystems which must be considered at a uniform level of detail, and also into subsystems which may be treated as behavioral units. A list of quantities and relationships of interest is extracted from the student's query. Using an ATMS,[7] Falkenhainer & Forbus find a minimal set of views which will support answering the query, and is consistent with respect to the dependencies between subsystems. These views are then used to generate the simulation, which is run and examined for the answer.

This work illustrates how constraints derived from the explanatory goal (i.e., what *must* be answered) and from reasoning requirements (what *can* be safely ignored) can be managed when choosing a simplified perspective for explanation-directed problem solving. Simplifications may also be chosen for pedagogical reasons, e.g. to enable the student to understand the explanation. Falkenhainer & Forbus do not address such constraints.

In summary, explanation is coupled to reasoning by choice of perspective in both directions. To the extent that a problem solving task requires using a given perspective, then the explainer must use some form of that perspective to accurately describe the reasoning. When alternate reasoning methods are available, then explanatory goals may direct the problem solver's use of perspective. In this case, the problem solver must be designed to be able to switch perspectives at the granularity required for the explanation.

3.2.4 The Relevance of Perspective to Explanation

I conclude with a summary of the reasons perspectives are relevant to explanation, and a brief discussion of implications for the theory and design of knowledge communication systems.

Conflicting Representational Requirements. At some point in a computational theory of reasoning, arbitrary choices are made in choosing algorithms operating on some data structures. It

[7] Assumption-based Truth Maintenance System, which simultaneously computes all sets of assumptions under which an assertion holds.

may be that the abstract model of reasoning underconstrains some of the choices made at the implementational level, or efficiency considerations may require that procedural knowledge be expressed in a form different than would be optimal for some communications. This suggests that distinct perspectives for machines and humans are sometimes needed in knowledge-based systems, and hence that an explanation facility will have to be able to distinguish and translate between the two.

Dealing with Complexity. In complex problem solving, it is at times expedient to utilize a simplified model for the sake of computational tractability. Using hierarchically organized perspectives, approximate reasoning at a coarser granularity can provide guidance for finer-grained search. Similar techniques can be useful in explanation to reduce complexity in ways which increase comprehensibility. Also, qualitative perspectives can be used in explanation to index quantitative information and organize it into meaningful categories. A complete theory of explanation will account for the use of perspectives as tools for communication.

Multi-Dimensionality of Understanding. Multiple perspectives are useful in most (perhaps all) domains of knowledge, because each enables people to do a kind of reasoning the other does not support. Consider the topological and synchronous models of electric circuits, or the wave and particle theories of light. Even when two perspectives are logically redundant with respect to each other, or one subsumes the other, an understanding of both and especially the ability to translate between them may be required before someone is said to fully understand the domain. Hence an explainer intended to aid in understanding such a domain will benefit from the ability to select appropriate perspectives and translate between them.

Constructing Understanding. In explaining or teaching, one must relate the content to be communicated to what the recipient knows, so that he or she may reconstruct the new knowledge in the context of his or her current knowledge state. Since people have different backgrounds, it is advantageous to be able to express the knowledge in different ways, making it possible to choose one which matches what the hearer can comprehend. Analogies to phenomena understood by the hearer are useful in making this connection. Also, people can't assimilate large bodies of knowledge at once: at times simplified intermediate versions are needed, which may amount to different perspectives. These final points indicate that the use of perspectives is tied strongly with pedagogical exploitation of epistemological structure, which I address in the next chapter.

Implications for Knowledge Communication Systems. While no one disagrees that people look at the world from multiple perspectives, Stevens & Collins (1980) conclude that there are strong implications for both expert system design and ICAI systems:

> "Our proposal is that expert systems need multiple models that can be used generatively to test out novel hypotheses and make predictions about new situations. Furthermore, they must have specific strategies to determine when to invoke one model and when another, and how to map back and forth between models. In sum, representation of expert knowledge must be further removed from the surface forms in which people talk than most current systems contemplate. ... This view suggests that multiple models should be taught explicitly as alternative points of view about a topic. The emphasis should be on the kinds of situations and problems for which each model is applicable, and on how to apply them to solve different types of novel problems."

Recall how MYCIN's performance rules, derived from the ways in which a physician speaks about diagnosis, proved too shallow a representation for the knowledge to be communicated by GUIDON. Similarly, Stevens & Collins argue that the scripts of the original WHY system were artifacts of how people describe the critical events occurring in the mental simulation models these researchers believe people use to understand physical processes. Regardless of whether their psychological claim is correct, I believe they, like Clancey, are right in advocating analysis, representation, and use of the more generally applicable, deeper models of a body of knowledge as the basis for explanation. Such an approach separates the essential knowledge from the form in which it is presented, enabling development of explanatory theory capable of choosing this presentation to be appropriate for the explanatory goals and recipient. Hence research on reasoning and explaining with "deep" models continues the trend towards an explicit theory of knowledge communication started in Carbonell's separation of knowledge from control in SCHOLAR (section 2.1.1).

Chapter 4

Explanatory Structure

Given the potential content of an explanation has been represented or otherwise made accessible, one must specify how that content is selected and organized in an explanation. Several levels of description are available within which one can articulate organizing principles. Research concerned with the logical structure of explanation was reviewed in section 2.3. There the concerns were with structural issues such as stating reasons before conclusions, limiting the depth of reasoning chains, and ensuring subproofs are nested. In this chapter, I review research emphasizing the epistemological and rhetorical structure of explanations. At an epistemological level of analysis, one is most concerned with ensuring that the content and expression of an explanation can be understood with an interlocutor's existing knowledge, or helps the interlocutor extend his or her knowledge in ways which people find natural. Rhetorical analyses of explanations have been concerned with describing patterns of explanatory text which humans find convincing for a variety of reasons, resulting in a mix of logical, epistemological, and linguistic concerns with "purely rhetorical" techniques to be discussed. It is difficult to draw clear lines between these levels, though an attempt is made in section 6.1. Underlying the material discussed here is a quest for an appropriate vocabulary for describing explanatory content and structure, and for the principles by which explanations are organized.

4.1 Epistemological Structure

In this section, I gather together research relevant to a computational theory of how an explainer structures an explanation to make contact with the interlocutor's current knowledge state, enabling him or her to comprehend it and integrate the new knowledge it communicates. That is, the work of this section examines the structure of explanation at an **epistemological**[1] level of analysis, asking how an explainer exploits the structure of knowledge for explanatory purposes. In particular, the material of this section examines how explanations are constrained by the internal structure of a

[1]See footnote page 23 for a definition of "epistemology". I use the term to highlight the emphasis of this section on theories of the nature of knowledge as the basis for understanding the structure of explanation. I acknowledge that I am borrowing a term from the philosophers, yet do not attempt to invent a new term because the fit is so close, and there is enough idiosyncratic jargon in my field as it is. I do not use the term "pedagogical" since it is too inclusive, including rhetorical, attentional, and motivational considerations as well as those having to do with the nature of knowledge itself.

domain's body of knowledge, the role of an individual's knowledge in understanding new concepts and situations, and the ways in which individuals are willing or able to transform these knowledge structures. Clancey's explanation by abstraction to familiar principles (page 25) is an example. By showing how a very specific rule is an instance of a general and accepted principle via subset and instance relations, such explanations both make contact with existing knowledge and help the hearer construct links from that to the new knowledge using relationships between knowledge which are also familiar. Recall also the use of hierarchical scripts in WHY (page 8) to progressively refine a student's understanding. The underlying principle is that each explanation should demand only incremental modifications to the student's model of the process being explained.

An influence on the work reviewed here is Piaget's **genetic epistemology**, a product of the **constructivist** school of thought (Furth, 1969; Piaget, 1971; von Glaserfeld, 1989). In this view, understanding occurs when information is assimilated to existing knowledge structures; and learning occurs when the agent generates new structures needed to accomodate to genuinely new information, and integrates these with existing structures. The implication for explanation is that people won't understand what you say, let alone be able to retain, retrieve, and use it, if you don't make some contact with what they already know. Furthermore, there are restricted ways in which people are willing or able to effect major transformations of the conceptualizations by which they understand the world. For explanation, this means the explainer needs to identify pedagogically exploitable relations between the to-be-communicated and whatever guesses are available about the hearer's knowledge state, and structure the explanation to follow these **epistemological gradients** (my phrase) along which he or she is likely to extend or reorganize his or her knowledge.

The line between short range explanatory dialogue and longer term curriculum design issues begins to blur in this section, since I will be considering the structure of extended sequences of explanations as well as of single explanations in a fixed context. However, I consider only work within artificial intelligence, ignoring much relevant literature from disciplines such as educational psychology, e.g. Hewson's (1981) theory of the conditions for "conceptual change". As in section 3.2, this section gathers research from diverse sources which is not the product of a single interacting subcommunity of researchers.

4.1.1 Exploiting the Structure of Knowledge

Here I examine three manifestations of the idea that one can follow the structure of knowledge when deciding how to select and order the content of an explanation. They differ on what "structure" is exploited.

Genetic Graphs. One of the first attempts at a computational theory of tutoring which takes into account the evolutionary nature of human knowledge was Goldstein's (1979) **genetic graph**. In this approach, knowledge was modeled as a set of rules organized in a network of links representing relations between rules, including generalization, specialization, refinement, and simplification. These relations were defined in terms of straightforward syntactic transformations between the expressions representing the rules. The genetic graph was used in two ways. First, it provided a basis for choosing the next topic for tutoring. Using the heuristic that learning is best done incrementally, building on existing skills, the candidate topics are those rules at the "frontier" of the overlay student model. Frontier rules are derivable from existing knowledge via a single transformation of one of the types previously listed. Second, once a target topic was chosen, the genetic graph supported explaining it

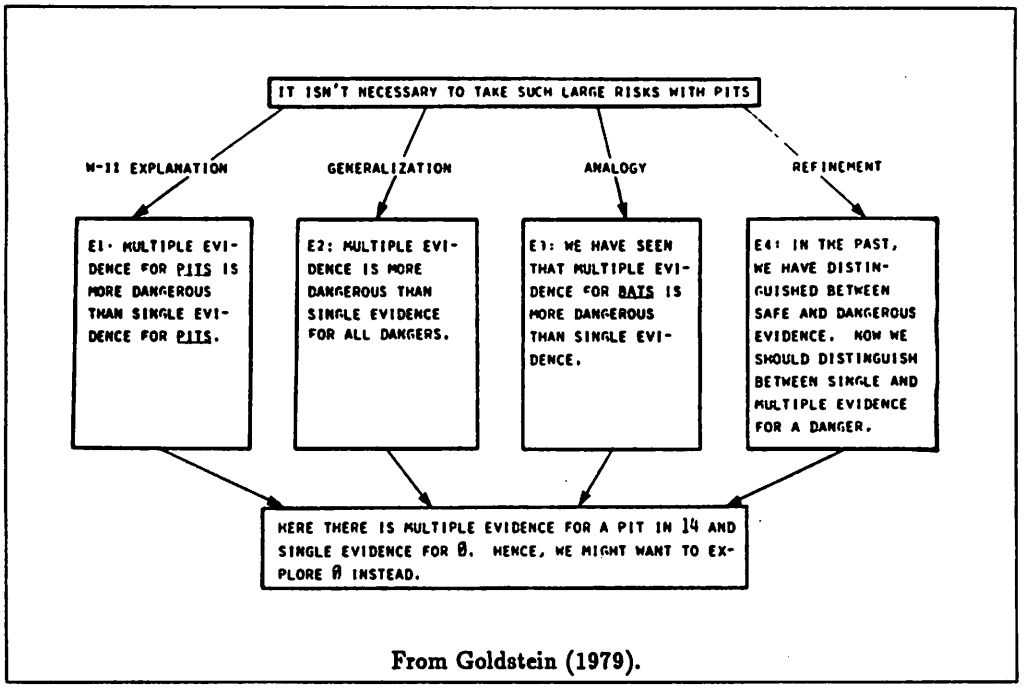


Figure 4.1: Explanation Types in the Genetic Graph

in multiple ways, to the extent that multiple links of different types could be found from it to topics already in the student model (figure 4.1). Each link type provided a distinct way to present the new topic in terms of its genesis from a topic the student has understood.

Genetic graphs are most applicable to fine grained, uniformly represented knowledge units, though the idea need not be limited to rules. The burden of constructing a large graph may be eliminated by automating this step (as Winkels, Breuker, & Sandberg, 1988 claim to do), and by computing neighborhoods of the graph only as they are needed. Syntactic transformations between units are convenient for computations such as this, but their appropriateness as the right gradients along which to guide human learning should be further evaluated.

Relations for Ordering Explanations. Bienkowski (1986) argues that "where natural ordering exists for a body of knowledge, it should be used ... a natural ordering is one that corresponds to some commonsense use of the knowledge". In contexts where the interlocutor will apply the knowledge in well defined and straightforward ways, this advice is unproblematic: one structures the explanation to follow the structure of the activity, introducing information at the point at which it is needed. However, explanation also includes communication of knowledge not directly linked to an activity, or which bears on that activity in ways one would not call common sense. In such cases, something other than an activity must serve as the source of ordering. Bienkowski provides examples of three kinds of orderings:

"Exploited": Explicitly represented orderings, including execution traces or operations of a theorem prover, support connections among assertions in a TMS, causal links, abstraction hierarchies, and goal refinement.

"Imposed": Relying on some knowledge source to find a useful ordering (not explicitly represented) in the knowledge. Examples given are the tour approach to describing apartment layouts (Linde, 1974), and the use of envisionment to find causal chains in a qualitative model (section 3.2.2).

"Derived": Getting the ordering from a reasoner that uses the knowledge (in some order) for some other purpose, or can reason about connections between knowledge.

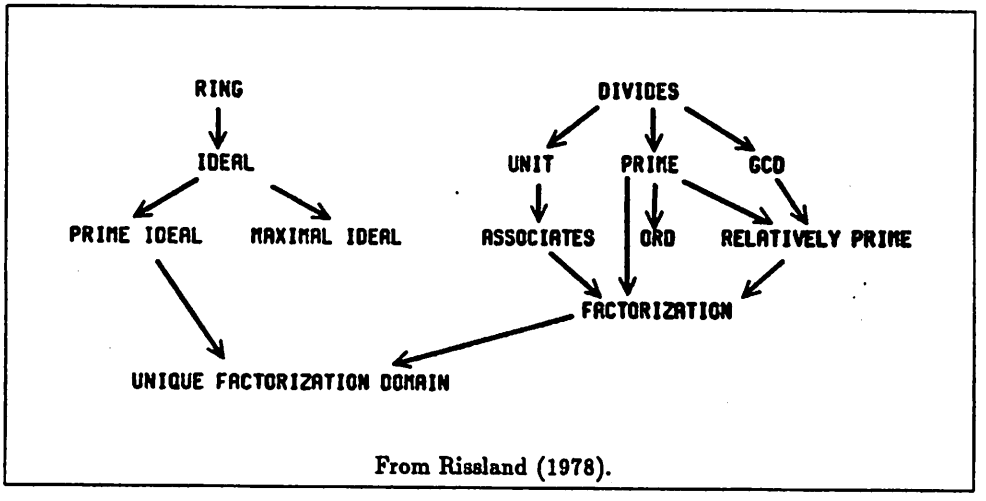
I have already given examples of "exploited" orderings which rely on the operations of a theorem prover in Chester's EXPOUND (section 2.3.1), on the support relations of a TMS in Weiner's BLAH (section 2.3.2), and on goal refinement in Swartout's XPLAIN (section 3.1.2). The use of schemata in McKeown's TEXT (section 5.1.1) is the most obvious example of an "imposed" ordering. However, the distinction between these types of orderings are not clearly defined. Consider an algorithm that traverses an explicitly represented partial ordering to derive a full ordering, and reasons about (say) connections to a model of interlocutor beliefs to make its choices. One could argue for placing it in any of the three types.

From a theoretical standpoint it would be preferable to avoid distinctions which are sensitive to the implementation or representation of knowledge, and abstract away to the epistemological basis for deriving explanatory structure. The above examples suggest that this basis resides partially in *conceptual relations and structures*, i.e. structures which convey propositional content. An explainer can exploit the ways in which these relations and structures connect concepts to each other, in an attempt to facilitate the interlocutor's reconstruction of the conceptual structures. To exemplify this informally:

- one describes events in causal order because people naturally infer causality from events perceived in time, and so will find it easier to organize descriptions of events in a causal structure if they are so ordered;
- one describes physical structure such as an apartment layout as if taking a tour because the hearer can then reconstruct the structure as a coherent whole, each new part being connected directly to this whole rather than being left "floating in space"; and
- one describes the assembly of an object in an order derived from traversing the goal structure of an assembly plan, because this helps people reconstruct the plan or goal structure, enabling them to recall each of the steps as they are needed.

In general, to choose and order the contents of an explanation in a way which is natural for the knowledge being communicated, an explainer needs to determine what kind of conceptual structure is appropriate, and know how to exploit it, regardless of how it is represented or accessed.

The Structure of Mathematical Knowledge. In an analysis of the epistemology of mathematics, Rissland (1978a,b) describes three categories of "items" around which mathematical knowledge is organized. **Results** are the logical-deductive elements such as theorems; **examples** are domain entities which illustrate or motivate other items; and **concepts** include definitions and heuristic advice. The items in each of these categories may be related in certain ways, resulting in **spaces** (which may be seen as perspectives on the domain of mathematics). Deductive relations give us the results space. Examples are organized by derivation relations showing how they are constructed from each other. The concepts space is organized according to a pedagogical prerequisite relation encoding both "expository tastes" and how concepts enter into the definition of other concepts. For



From Rissland (1978).

Figure 4.2: Concepts Space for Unique Factorization Domain

example, figure 4.2 shows one way to structure the concepts behind that of a "unique factorization domain".

Rissland notes that the relations between items are as important to an epistemology of the domain as the items themselves, and provides further structure using inter-space **dual relations**. For instance, the pre-duals of an example in concept space are those concepts required to construct the example, and the post-duals of the example in concepts space are the concepts motivated by it. The post-dual of a result in examples space may be an example which the result does not apply to. Dual relations are similarly defined between other combinations of results, examples, and concepts. Duals also provide a means of relating two items which reside in the same space, even though they may seem poorly related by intra-space relations. Such items are said to be **close in the dual sense**. For example, two results may have distinct deductive lineage, yet both apply to many of the same examples; and two concepts may be independently definable, yet be shown by a result to be identical or strongly related. These relations provide indices which may be useful in retrieving material relevant to the topic of an explanation.

Items in each space are further categorized into **epistemological classes** according to the roles they play in the development of the mathematical theory as a whole, as determined by examination of the dual relations. For instance, easily understood examples used as initial illustrations of concepts or results are **start-up examples**; those which motivate further development of the theory by limiting results are called **counter examples**; and **reference examples** are used throughout the theory to connect and test a variety of ideas. Similarly, there are **basic results**, **transitional results**, and **key results**, among others. Rissland (1978a) provides many examples of these. Her discussion is oriented more towards using these classes to organize a body of knowledge, rather than to their use in deciding what to include in an explanation. However, the epistemological classes are candidate abstractions for writing a general theory of explanation.

Items also have **settings**, i.e. the domain within which an example is given, a concept defined, or a result proven. In mathematics, these correspond to systems such as the integers or real numbers. It is necessary to include the setting in any representation of an item. Rissland discusses other requirements for an item to be **well-stated**, and further details the representation of items in each

class. Similar considerations concerning minimal content may apply to the well-formedness of an explanation.

Overall, Rissland's work is presented as a framework for the analysis of a body of knowledge, in particular for "understanding understanding" in the domain of mathematics. She expects this framework to be useful to help students and practitioners of the domain organize their knowledge and explorations, and has built an interactive computer-based system for facilitating the process. Further refinement of the framework is possible and needed. Rissland's undifferentiated pedagogical relation in the concepts space is a compilation of the epistemic and rhetoric considerations which motivate choice of ordering. Other ways of structuring examples space such as analogy may be useful. However, this impressive attempt to analyze the epistemology of a complex domain provides ideas (such as the pre- and post-duals) applicable to analyses of other domains, and shows the importance of sometimes neglected aspects of our knowledge, such as examples, in the genesis of understanding. Her analysis suggests that explanations can be planned to mirror the historical genesis of a body of knowledge, an intriguing "ontogeny recapitulates phylogeny" analogue which is controversial among educators concerned with curriculum design.

Summary. The intuitions that explanations should follow a natural order and that learning occurs along incremental gradients have led to examinations of the structure of knowledge as a source of constraints on appropriate choice and ordering of explanatory content. The work differs in what kind of structure is exploited. The genetic graph suggests that fine grained transformational relations between the *representations* of different versions of skills or concepts can guide generation of explanations aimed at refining these skills or concepts, though this approach may be limited by its reliance on syntactic transformations between representations. Conceptual relations, whether explicitly represented or implicit in a knowledge base, are an appropriate source of constraints on ordering when one wants to facilitate reconstruction of these relations in the interlocutor's mind. Finally, at a larger granularity, the logical and historical evolution of a body of knowledge provides a variety of relations between concepts, examples, and results which may be used to choose relevant material and order extended explanations.

4.1.2 The Role of Examples in Explanation

Two characteristics of examples make them important to explanation. Being objects of direct experience, examples are more memorable than abstractions, and hence are especially useful through the dual relations as indices to aid memory of concepts, heuristics, and results. Furthermore, examples also provide guidance in structuring one's communications. In the "ubiquitous dialectic" (Rissland, 1984a), examples delineate knowledge, by showing the limits of generalizations and motivating construction of more adequate ones. Bringing these observations together, knowledge motivated by a memorable example is more likely to be integrated with existing knowledge. This means that the explainer should consider what examples are available for motivation and indexing in deciding what to say next. In this way, examples themselves impose epistemological structure on a domain (Rissland, 1978a,b).

Constrained Example Generation. Rissland (1980) provided a framework for indexing, retrieving, modifying, and constructing examples based on constraints. This was applied to an on-line VMS

help facility in Rissland, Valcarce, & Ashley (1984). A few heuristics for the use of examples were given. Unfortunately this work was limited by a schematic approach: "TEXPLATES" combined text with specifications of which examples to use, or constraints on their generation (e.g. to use recently mentioned concepts in the example). In the next chapter I will discuss some of the limitations of schemata, which also apply here: the machine doesn't reason in a principled manner about the features of an example by which it supports one's explanatory goals. Hence the TEXPLATES implement but don't illuminate the epistemological roles of examples. Subsequently, Suthers and Rissland (1988) improved on the example generator by adding constraint satisfaction and optimization techniques, and a hierarchy of features to permit abstraction of pedagogical utility from the surface features of examples. Such abstractions may support a better interface of an example generator to explanation planning.

Research in inductive learning which emphasizes the role of the teacher in choosing examples provides insights into the use of examples as a basis of communication. Without attempting to cover this literature, I examine two major research efforts in this direction.

Near Misses. Winston's (1975) work on learning structural descriptions from examples provides a simple example of an epistemological principle applied to pedagogy, namely that one should make it possible for the learner to modify his or her knowledge in small increments. In his domain, the teacher's task was to come up with a sequence of examples which facilitated learning the concept of an arch. Special importance was attached to "near misses", counter-examples which fail to be instances of the concept, but not by much. The crucial aspect of these examples was that they differed from the machine's current conceptualization in a small number of ways (preferably one), and hence localized the change that needed to be made. Winston relied heavily on having access to the learner's concept representation.

Felicity Conditions for Inductive Learning. More recently, VanLehn (1987) undertook a careful analysis of implicit conventions for choosing and ordering examples in inductive learning. In his domain, learning subtraction from lesson sequences, the student infers a procedure for subtraction by observing the steps the teacher takes in solving problems. The student is assumed to be incrementally modifying a representation of a procedure, and the teacher's goal is to present lessons which bring this representation towards the target procedure.

When teacher and learner are mutually aware of their involvement in an inductive learning situation, implicit conventions for interpreting lesson sequences apply. These are analyzed and expressed in vanLehn's theory as **felicity conditions**[2] on inductive learning:

Minimal Set of Examples: Enough examples must be presented so that, in principle, the concept can be learned.

One Disjunction per Lesson: A disjunction is a conditionalization of choice of action on some variable. Disjunctions cause the hypothesis space to grow rapidly if they are not controlled. Just as Winston restricted modification of the learner's concept of an arch to one feature at a time, vanLehn restricts modification of the learner's procedure to adding one new branching at a time.

[2]The name was chosen to point out the similarity to language conventions described by Austin (1962).

Show All Work: If intermediate steps are permitted without the teacher showing them, there is potentially an infinite number of procedures possible depending on what these steps are. This condition applies only to "normal" lessons; "optimization" lessons may teach how to make use of "hidden objects" to make procedures more efficient.

Given the teacher and learner implicitly agree that these conventions govern the lesson sequence, the learner can rule out a large number of changes to his or her representation of the procedure, and concentrate on a small number of incremental modifications. These conventions would have to be modified somewhat to apply to the use of examples mixed with other forms of explanation. For example, declarative propositional communications eliminate the Minimal Set requirement, though it reappears as a general requirement that the explanation meet the interlocutor's informative needs. The One Disjunction requirement could still apply, and indeed is a specialization of the general principle that explanations should only challenge the interlocutor's knowledge in small increments.

Little work in AI has focused specifically on the use of examples in explanation. The educational literature is stronger in this area, though I have not reviewed it: see for example Tennyson & Park (1980). Many explanation systems make use of examples, but these are usually hard-wired in, and no theory of their use or impact on knowledge communication is provided. The usual mistake is to see examples as only illustrative embellishment on the content of an explanation, rather than also as signposts and junctions in its epistemological structure.

4.1.3 Accounting for User Goals and Beliefs

In this section, I discuss a few ways in which a model of the interlocutor's knowledge, beliefs, and goals could be taken into account in planning the content and presentation of an explanation. In particular, a **user model** can be used to determine what the interlocutor will understand, identify missing knowledge which should be communicated, and control inferences the interlocutor is likely to make. I do not attempt to review the literature on constructing user models. The interested reader is referred to Brown & Burton (1978) for inferring deep misconceptions from student's procedural errors; Rich (1983) for the use of personality stereotypes; Carberry (1983) for goal inference techniques; Kass & Finin (1987, 1988) for covert acquisition of a user model using stereotypical defaults and inference rules; Wahlster & Kobsa (1986) for a review of dialogue-based user modeling; and to Clancey (1986b) for a review of qualitative student models in instructional programs.

Using User Types. One of the simplest user modeling techniques is to categorize people into user types. One can then characterize what kind of information each type of user is likely to be interested in, and present a given user with only the information meeting this characterization. For example, recall the "viewpoint" mechanism of Swartout's XPLAIN (section 3.1.2), which labeled problem solving steps as to whether they would be of interest to computer specialists and/or medical audiences, and filtered them on this basis. Such broad categories implicitly select and filter content based on both the *presumed goals* and *presumed background knowledge* of the user group. Finer-grained content selection requires correspondingly finer-grained information about the user; and more sophisticated techniques treat goals and knowledge differently. Goals in a user model allow one to determine what additional information the interlocutor may need to achieve these goals, beyond that which was explicitly asked for. A model of background knowledge indicates concepts and terminology the interlocutor is likely to be familiar with, and what knowledge the interlocutor lacks.

Determining what the Interlocutor Understands. There are two reasons for doing this: to ensure that the primitives used to explain something new are indeed primitive, i.e., already understood by the interlocutor; and to leave out of the explanation unnecessary information which the interlocutor already knows. XPLAIN's filtering illustrated a simple attempt to ensure one uses terminology which the user is familiar with. White & Frederiksen (1987) suggest that one should select a perspective based on student background, e.g. explaining the functioning of an electric circuit using mechanical concepts to an engineering student, but using cybernetic concepts to one with a background in computer science. At a finer granularity, the notion of **prerequisite** so common in tutoring systems applies here: one could check whether the concepts relied on to explain a goal concept are present in the user model, and generate subgoals to first explain those which are not.

Models of the user's beliefs may be used to filter from an explanation content which the user is believed to already know. This was illustrated by Weiner's (1980) BLAH (section 2.3.2), which pruned from a justification tree assertions provable from a user's "view". Cohen (1988) and Cohen & Jones (1987) combine such deletion of user background with addition of information necessary to meet the user's goals. Filtering unnecessary information also indirectly focuses the communication on what the user may *need* to know, the other half of Grice's maxim of Quantity (section 2.2).

Identifying Knowledge the Interlocutor Needs. A user model containing goals and beliefs helps an explainer determine that it is appropriate to communicate information or knowledge other than what the interlocutor requested directly. The simplest means of doing so is illustrated by the overlay student model (section 2.1.1): candidate topics for tutoring are those missing from the student model. One may also generate informative goals by observing a *discrepancy* between the user model and the explainer's own beliefs. This is illustrated by the genetic graph (section 4.1.1), and also by McCoy's (1985, 1986) work on responding to misconceptions, discussed in section 5.1.2. Finally, if one knows the interlocutor's goals and some possible plans or actions for achieving them, one can identify and supply additional information which is needed to enable the interlocutor to carry out these plans (Allen & Perrault, 1980; Carberry, 1983). To give a classic example, the traveler asking a ticket agent what time a flight leaves may receive directions to the gate as well as a time in response, especially if the time is nigh.

Exploiting and Controlling Inferences. Explanation is structured in part along inferential frameworks, and understanding inferences made as part of the process of understanding is necessary to understanding the epistemological structure of explanation. The utility is twofold: to increase the informativeness of one's explanation, and to prevent erroneous interpretations. In a paper on question answering, Lehnert (1984)[3] points out that if the speaker knows something about what inferences the listener can make, he or she can avoid saying obvious things and convey more information by setting up inferences. For example, when saying why something happened, one can give an answer one or two causal inferences away from the direct "reason" for the event: then the listener not only gets the information about the direct reason but also about indirect causes. Plausible inference techniques such as those investigated by Collins (section 2.1.1) are relevant to determining what the interlocutor will infer without effort, and hence what can be communicated implicitly in an explanation.

Another application of a user model capable of modeling inferences the hearer can make is to *prevent* the hearer from inferring something which is *false*. For example, if the user's query Q

[3]Which also contains other substantial additions to her 1977 papers, reviewed in section 2.2.3.

presupposes P, but P is false and a literal response to Q does not deny P, then a denial of P should be planned to avoid appearing to sanction it (Kaplan, 1983). Joshi, Webber, & Weischedel (1984) and vanBeek (1986, 1987) address two variants of this. One is to prevent an inference which, by default reasoning, is implied by something the explainer plans to say, but is not true in this particular case. The other has to do with expected informative behavior. If the hearer expects the explainer to inform him if P is true, then failure to do so would lead the hearer to infer that P is false. For example, consider the consequences if a medical expert system fails to mention an immediately life threatening condition, simply because it was not asked about that possibility. These techniques are means of satisfying Grice's maxim of Quality (section 2.2).

Final Comments. The above applications of a user model are not disjoint. For example, in section 5.1.4, I summarize how Paris (1985, 1987) switches between functional and structural descriptions about devices according to the user's level of expertise with respect to the current subtopic. While one could view this work as selecting, at a fine granularity, the conceptual framework the user is likely to understand, it really is an example of identifying needed information and exploiting inferences. The expert is able to infer function from structure, whereas the novice is not, and therefore needs to be explicitly told about function.

Many of the techniques discussed throughout this section on epistemological structure appear to depend on an unusually good model of the interlocutors's knowledge. One might be tempted to discard these techniques as unrealistic, given the difficulties involved in user modeling (Self, 1988). However, the potential utility of the techniques is so great it may be worthwhile to assume an incomplete and erroneous model is correct, plan an explanation on that basis, and then deal with retraction of assumptions should things not turn out so well. Humans don't have direct access to each other's mental states either, so plausibly do something similar. This underlines the importance of defaults and nonmonotonic models (Kass & Finin, 1987, 1988), and of reactive planning techniques (Moore & Swartout, 1989), discussed in section 5.2.3. Also, it is to an extent possible to exploit epistemological structure without a user model. An explainer can reason about the connections between knowledge to order explanatory content in a manner most people will find comprehensible, regardless of individual beliefs or goals.

4.1.4 Perspectives and Epistemological Structure

In this section I discuss how multiple perspectives, and related notions, figure into the problem of planning explanations at an epistemological level.

The Role of Assumptions in Explanation. DeKleer & Brown (1983) emphasize the importance of understanding the role of assumptions in the process of "envisioning" a causal model from a structural one (section 3.2.2). Careful use of assumptions also aids in explaining reasoning based on a potentially complex theory. The learner must construct his understanding incrementally from his initial knowledge state, so the explainer needs to communicate with simplified theories which can be composed by the learner into a complete theory, without the learner relying on that complete theory for integrating the simplified theories (recall the "T_0 problem", page 36). In particular, the explainer may do so by selectively violating the "no function in structure" principle, then discharging the assumptions made:

> "In explaining how a device works, one wants to construct a *sequence* of explanations, commencing with one built around component models that have the not-easily-understood aspects of the device's functioning implicitly embedded in them. That is, it is often pedagogically expedient to let an "explanation" presuppose part of what it is trying to explain. By using highly simplified primitives (component models), the correct causal model or running process can be more easily communicated. Futhermore, from the learner's perspective, the simply constructed, but correct causal model can serve as a cognitive framework for organizing forthcoming refinements ..." (DeKleer & Brown 1983, p. 184)

DeKleer & Brown also suggest that if the first causal model is chosen carefully, the refinement will only add to it, without ever having to radically reformulate its organization. They acknowledge the possibility that a model meeting this requirement may be too complex, and suggest using a hierarchy of models based on changing resolution and embedded component models to permit choice of a simple enough initial model. Their theoretical framework has provided an indication of the direction elaboration proceeds in, but further work is required on knowing when to retract a given kind of assumption, or how to move within model hierarchies. The next body of work attempts to address these issues.

Model Progression. White and Frederiksen are developing a pedagogical theory of model progression in teaching electrical network theory, based on three dimensions on which models[4] vary (Frederiksen & White, 1988; White & Frederiksen, 1987). The **perspective** dimension is concerned with "the nature of the model's reasoning", and includes functional models of the purpose of the circuit and how components interact to achieve that purpose; behavioral models which describe how changes in the state of one component causes changes in the states of others; and reductionist physical models, where behavior of a component is derived from its molecular structure The **order** dimension describes quality of information, increasing from zero-order models reasoning on the basis of presence or absence of some quantity, through first-order models, which reason about incremental changes in quantities, to quantitative models. The **degree of elaboration** of a model is defined in terms of the number of constraints considered in causal simulation.

These dimensions implicitly define a space of models with transformations between them. In electrical network theory, for example, pedagogical model evolution may occur along the elaboration dimension by adding first knowledge and principles enabling reasoning about voltage distributions, and then principles for reasoning about current flow. Each of these clusters of knowledge may first be presented in their zero-order form, being concerned only with existence of quantities and properties, and then transformed to their first-order and quantitative forms. Perspective transformations may occur from predominately causal-behavioral to a functional perspective, to develop an understanding of the purpose of circuits and circuit components, and/or to a reductionistic, physical perspective in which microscopic phenomenon are used to derive the previously assumed rules governing the behavior of components. In addition to these cross-dimensional transformations, **micro model evolutions** may occur *within* a given Perspective and Order[5] via incremental changes similar to those described in Stevens & Collins (1980), and also to those used in Goldstein's genetic graphs. This includes **acquisition** of a new concept, law, or problem solving skill; **differentiation** of an existing concept into two concepts; **integration** of two concepts which are related; **generalization** of

[4] Perspective, in the terminology of section 3.2.

[5] And within Elaboration, according to White & Frederiksen, though micro model evolutions seem to me to constitute a change in Elaboration.

a concept to apply in a wider range of contexts; **refinement** of an existing concept; and **substitution** of one concept or skill for another.

The papers provide detailed examples of the use of these models and model evolution transformations between them. Most of the discussion is limited to what has been implemented, namely zero-order models and a linear progression of increasing elaboration. White & Frederiksen present what they claim to be an empirically good curriculum for the circuits domain, but only minimal principles for deriving model progressions in other domains.

4.1.5 Coherence and Consistency Issues

Thagard (1988) is concerned with evaluating competing scientific explanations on the basis of their **coherence**. Though he is discussing "explanation" in its sense as a relation between scientific theory and observations rather than as a process of communication, coherence considerations also apply to the latter. Thagard takes "explains" as a primitive relation between propositions, and defines coherence in terms of this relation. To summarize informally, a scientific theory is more coherent (and hence preferred) to the extent that its propositions are connected to each other and to the observations by explanatory relations. Thagard's use of explanation as a primitive notion limits the usefulness of his work for the purpose of understanding the coherence of explanatory communications. Perhaps this may be dealt with by using other relations, e.g. causal ones, as the primitives. In spite of this, the corresponding intuition for explanation is clear: an explanation will be better accepted if it is internally coherent, and easier to integrate into existing knowledge if it is externally coherent (i.e., with that knowledge).

There are consistency issues in the use of multiple models or perspectives as well. White & Frederiksen (1987) discuss the need to maintain **causal consistency** between models. For example, most qualitative causal models of electricity treat voltage as primary and current as its consequent (provided resistance is not infinite); yet quantitative constraint models use algebraic variants of Ohm's Law such as $V = IR$, which inconsistently implies that voltage is a causal consequence of current. The coherence of a curriculum (and also of an explanation) is improved if gratuitous inconsistencies between different types of perspectives are avoided. Consistency also bears on the relation between simplified and refined models. Recall DeKleer & Brown's suggestion that incremental model refinement is possible. White & Frederiksen make a similar argument for **modifiability** or **upward compatibility** — each model should be constructed such that it can be extended or refined to better serve the intended purpose with minimum revision to existing conceptions. The ideal is a model progression in which the initial models are not simplified at the expense of introducing misconceptions (which must be undone when presenting subsequent models). While this is an ideal worth striving for, I suspect it is a rare domain in which simplified explanations will not engender misconceptions. Furthermore, misconceptions and inconsistencies may be *pedagogically necessary* to create enough "cognitive dissonance" to motivate learning. Also, people don't come without misconceptions, so a serious effort towards user modeling will require representing and reasoning about faulty and inconsistent models.

A related but distinct issue is whether the variety of perspectives one can take on a domain will be consistent in the explanatory moves they recommend. Lesgold (1988) builds a **curriculum lattice** by intersecting four hierarchies which index topics in electrical network theory under four corresponding "viewpoints": Circuit Types, Laws, Electrical Concepts, and Problem Types. Surprisingly, Lesgold finds that all four viewpoints are consistent in their constraints on how to sequence the curriculum. It

seems likely that a more generally applicable theory of explanation will have to include some means of conflict resolution between the different recommendations of ordering relations over the content to be covered.

I believe that epistemological considerations are responsible for some of the structure attributed to "rhetoric" in the work I discuss next. Indeed, many algorithms and heuristics for explanation appear to be implicitly based on constraints generated by the need to make contact with and build on the interlocutor's existing knowledge. Existing explanation theory needs something analogous to what Clancey did for MYCIN (section 3.1.1): explanatory principles or strategies should be stated explicitly at an abstract level, with the connection to specific knowledge and situations retained via appropriate abstractions. Research addressing the epistemological constraints on explanation has been scattered. A better understanding of the epistemological structure of explanation is essential to provide a theoretical justification of the success of existing algorithms and heuristics, and improve on their failures. Such an understanding may have additional applications outside of explanation, in areas such as machine learning and knowledge acquisition, and knowledge communication between distributed agents. Hence, identification of the epistemological principles of knowledge communication and their operationalization within a unified computational framework is an important research problem, and one which I intend to address in my own work.

4.2 Rhetoric and Argumentation

The material of this section has in common an emphasis on the persuasive impact of an explanation or argument, and adds at least one more dimension of explanatory structure (the rhetorical) which a full theory of explanation must account for.

4.2.1 Rhetorical Structure Theory

Here I discuss research concerned with three closely related notions: rhetorical structure, rhetorical relations, and rhetorical predicates. To preview, the rhetorical structure of a text may be described in terms of functional relations between text components, and a generative theory based on these ideas uses predicates to select content which fills the desired roles (section 5.1.1).

Rhetorical Relations. Although the idea of rhetorical relations may be attributed to writers as far back as Aristotle, McKeown's dissertation (1982) was the strongest direct influence of rhetorical considerations on computational theories. In McKeown's words (she uses the term "predicate" to emphasize her use of the relations for selecting content meeting a functional role):

> Rhetorical Predicates are the *means* which a speaker has for describing information. They characterize the different types of predicating acts he may use and delineate the structural relation between propositions in a text. (McKeown 1982, p. 22).

Figure 4.3 lists all the relations used in her analyses, grouped under their sources, as well as some added by Mann & Thompson (1983, to be discussed below). McKeown defines her relations by example. There are overlaps, for example, Specification, General Illustration, and Particular Illustration

From Grimes. (As presented by McKeown.)	
Attributive	Predicating a property.
Equivalent	'When we say $P(x)$ we mean $Q(x)$ and $R(x)$.'
Specification	'N. In fact, S.'; S more specific than N
Explanation	Reasoning behind an inference drawn.
Evidence	'N. S.'; S is evidence for N.
Analogy	(McKeown's example is a similarity comparison.)
Representative	Item representative of a set is given. (Special kind of example.)
Constituency	Presentation of sub-parts or sub-classes.
Covariance	"Antecedent, consequent statement." (Why directional?)
Alternatives	A disjunction to choose from.
Cause-effect	'P causes Q.' (Mann & Thompson distinguish volitional/nonvolitional.)
Adversative	"It was a case of sink or swim."
Inference	Stating the inference: '... therefore P'
From Shepherd (As presented by McKeown; used to teach writing style).	
Topic	Introduces the topic.
Amplification	Expands on the topic (a.k.a. "Elaboration".)
General Illustration	Universal statement supporting claim.
Particular Illustration	Specific statements supporting claim.
Comparison	Noting features in common between two entities.
Contrasting	Noting feature differences.
Conclusion	
Added by McKeown.	
Identification	Saying what a term refers to. (Compare to Equivalent.)
Renaming	'N, also known as M.'
Positing	Putting forth as fact; to postulate. (A hypothetical?)
Added by Mann & Thompson (partial listing).	
Antithesis	S and N are contrasted, with committment to N.
Background	N cannot be comprehended without information in S.
Circumstance	S provides situation within which N is interpreted.
Concession	Acknowledgement of S, which potentially detracts from N.
Conditional	S provides condition under which N holds.
Enablement	N is a directive; S enables addressee to comply with it.
Motivation	N is a directive; S motivates complying with it.
Sequence	Succession of items along some dimension.
Solutionhood	S provides a solution to the problem posed by N.

Figure 4.3: Examples of Rhetorical Relations

are special kinds of Evidence. The relations describe textual roles and structure due to a variety of sources. Some correspond directly to individual relations one might store in a knowledge-base (e.g. Constituency), while others are about more complex relations between concepts which may vary according to the context (e.g. Comparison). Many of the relations are useful because of how they exploit epistemological structure in helping the hearer grasp a point (especially Background, but also Representative, for example). Illocutionary speech acts are involved in the relations as well (most blatantly, Concession; others are not themselves illocutionary acts but only make sense in the context of certain illocutionary acts). Finally, relations such as Topic and Conclusion appear to be due to conventions governing writing style, but may be justified by the characteristics of human focus of attention.

Rhetorical relations were an important development in explanation research, since they provided a first pass at abstractions for a general description of explanatory structure, and in bringing various roles of the parts of explanation into the foreground, point out phenomena in need of further study. However, researchers have nearly exhausted the direct utility of rhetorical relations for supporting further advances in a theoretical understanding of explanation.[6] Rhetorical relations fail to make the necessary distinctions for such advances because they are descriptive of explanatory *text*, i.e. the end product of some explanation generation process, and so describe with one device structure due to a variety of distinct knowledge sources and constraints bearing on such a process. What appears to be a single structure in the text may in fact have arisen from multiple constraints and heuristics. Further analysis is needed to identify relations which are conceptual abstractions, and those derivable from epistemological considerations such as those discussed in section 4.1, or from an understanding of human attentional limitations. It would be interesting to see what remains of the purely motivational or persuasive aspects of rhetoric, and what is merely due to stylistic conventions. Such distinctions correspond to aspects of communicative expertise which may operate on different knowledge structures (as illustrated in section 6.1), and hence need to be separated in a theory which is based on an analysis of the deep structure of explanation, rather than being descriptive of its surface structure.

Having introduced the relations themselves, I now illustrate their use in analyzing and describing text. I leave their use in a generative theory to section 5.1.1, where I describe McKeown's schematic theory of text generation.

Rhetorical Structure Theory. Rhetorical Structure Theory (RST: Mann, 1984; Mann & Thompson, 1983, 1986) was heavily influenced by McKeown (1982, 1985), in particular by her rhetorical predicates. To analyze a text using RST, one first breaks the text into units of convenient size (which may be single clauses, sentences, or paragraphs). Then one uses rhetorical relations between the units to combine them into larger units, called **rhetorical structures**. These consist of at least two text units, the **nucleus** and one or more **satellites**, plus the rhetorical relation which describes the function of the satellite with respect to the nucleus. The nucleus is always the more prominent, essential span of text, and the satellite the supporting material. The result of the analysis is a structural hierarchy based on a functional description of the parts of the text. An example of a rhetorical analysis, from Mann & Thompson (1986) is shown in figure 4.4.

Some of the rhetorical relations used in Mann & Thompson's RST are listed in figure 4.3, where N denotes the nucleus and S the satellite. They are developing semi-formal definitions of rhetori-

[6]This is my personal view; some researchers would no doubt strongly object.

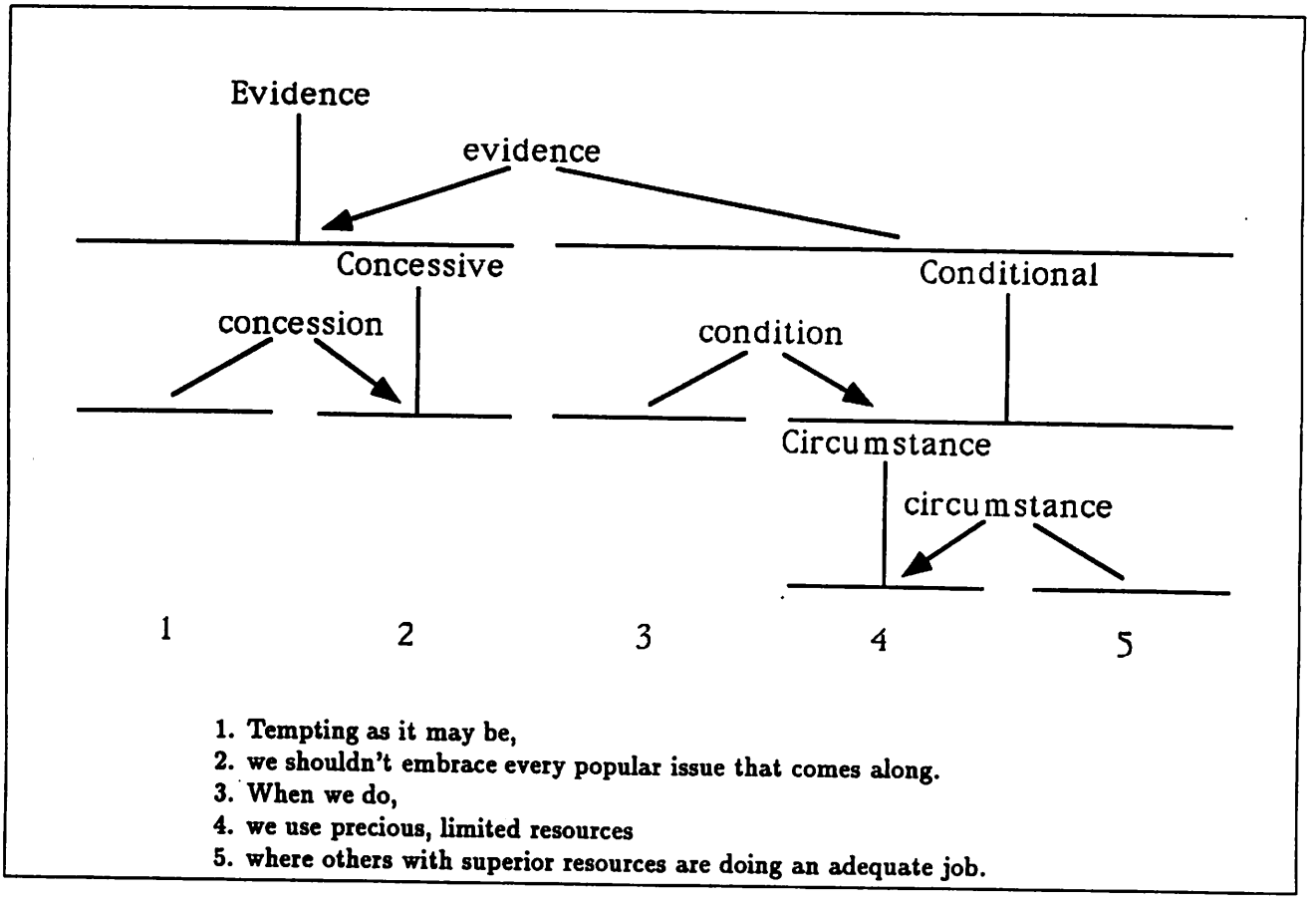


Figure 4.4: RST Analysis of Simple Text

cal relations which include various constraints and an effect, as illustrated below for the Evidence relation:

Constraints on the Nucleus: The reader possibly doesn't already believe the claim.

Constraints on the Satellite: The satellite being the evidence, the constraint is that the reader either believes it or will find it plausible (possibly due to rhetorical structure internal to the satellite).

Constraints on the Combination: Comprehending the evidence will increase the reader's belief in the claim.

The Effect: The reader's belief in the claim is increased.

Note that the effect may be seen both as the communicative goal and as an inference which the speaker makes of the hearer's belief after completion of the communication. RST was developed as a descriptive theory, and only recently is being used for generating text (Hovy, 1988a,b; see section 5.2), though McKeown did use a precursor of RST.

Relational Propositions. An important argument for considering rhetorical relations when planning explanations centers around the notion of **relational propositions** (Mann & Thompson, 1983; 1986). These are assertions which are not explicitly expressed in, yet are effectively communicated by the text. For example, we rarely say "*N*. *S*. *S* is evidence for *N*."; we more often say "*N*. *S*." and expect the hearer to infer the proposition that *S* is evidence for *N*. (In figure 4.4, *N* corresponds to 1-2 and *S* to 3-5.) Mann & Thompson claim that every relation in RST conveys some relational proposition, and that these propositions are essential to the coherence of the text. Implicit propositions are a means for achieving brevity, and text that conveys them explicitly could seem excessively repetitive. Hence, at some point in the generation of text from a language independent representation of the explanation, choices should be made concerning which propositions can be expressed implicitly in the rhetorical structure of the text. As noted previously, theoretically distinct considerations are mixed in the rhetorical relations. Once these considerations are separated, an explanation realizer may have to map a specification of explanatory structure at several levels of description into the implicit structure of the realization medium.

Coupling of Explanation to Text Planning. This leads to the problem of the relationship between media-independent explanation planning and its realization in a media such as text. How far can explanation planning proceed without considering the characteristics of the realization medium?[7] This is currently a matter of debate. There is general agreement that some constraints flow backwards from the media to content planning, but difference of opinion on how this should impact on a theory of explanation. To illustrate, the following approaches were discussed at the AAAI-88 Workshop on Text Planning. One is to model the interaction in the planning process. Cecile Paris combines content selection with rhetorical planning, and treats realization as a separate process. However, the text planner has "windows" on the realizer, through which it can ask questions concerning whether a plan fragment is realizable in a certain way. The problem with this approach is the danger of merging the two processes if there are not clear and restrictive criteria on the interface. It is certainly more elegant to design an architecture which *cannot* plan something that cannot be realized. Marie Meteer

[7]Relational propositions could occur in other media such as graphics — I have not examined the literature on this, and am unsure of whether the implicit vs. explicit distinction makes sense in graphical media.

is attempting this by using a series of intermediate languages for Conceptual, Linguistic, and Surface level plan specifications in a pipelined architecture. The design of each language is intended to ensure that messages expressed in it have the properties needed to guarantee that the next step can proceed. A third approach is to merge linguistic considerations with content planning, perhaps based on the standpoint that language is not just a surface phenomenon, but rather is deeply involved in thought processes. For example, Paul Jacobs encodes alternate lexical, conceptual, and speech act structures in the knowledge base, and selects them automatically according to context (as opposed to during text planning). In this literature review, I have opted to stay away from these complex issues, and (naively) treat explanation planning as if it could occur largely without the help of linguistic knowledge sources.

4.2.2 Argument Structure

Like RST, studies of argument structure provide a dimension of explanation not well covered by researchers in other traditions. As Stucky (1986) puts it: "Whereas the logician will be most concerned with the analysis of a single argument, the rhetorician will study the interaction of many arguments in determining their persuasive appeal."

The Atoms and Molecules of Argument Structure

The TRJ Model. In argumentation as well as explanation, it is not sufficient to simply apply rules of inference. One must argue for the use of the rule itself if one is to be convincing. Toulmin (Toulmin, 1958; Toulmin, Rieke, & Janik, 1979) elaborated on the structure of simple inference rules to support this. As illustrated in figure 4.5, the **warrant** is the unelaborated rule; the **grounds** are the contents of working memory which meet the antecedent of the rule, and the **claims** are the instantiated conclusion of the rule. He added the **backing**, which supports the credibility or correctness of the warrant, and **rebuttals** which acknowledge potential ways in which the claims may be attacked and defends against them by limiting the scope of the claim. The schema of figure 4.5 has aided some researchers in the task of argument analysis, and variants of the TRJ model have influenced some recent work in explanation, eg. Rochowiak (1988) and Wick & Thompson (1988). However, the model is limited in saying little about backings and rebuttals, other than that they impact on the credibility of inferences. In contrast, the work discussed in section 3.1 takes the stance that backing in diagnostic domains consists of abstract strategies and the domain knowledge by which these are instantiated into inference rules. Rebuttals may be better understood in terms of the role of assumptions in limiting the applicability of models for reasoning (section 3.2.2).

Argument Molecules. "Argument graphs" (Birnbaum, Flowers, & McGuire, 1980) show support and attack relations in a manner similar to the TRJ model. Birnbaum (1982) describes tactical "argument molecules", which are "patterns of support and attack relations that encompass several propositions". These express implicit structural relations which play a functional role in understanding and generating arguments, in a manner similar to the "relational propositions" of section 4.2.1. For example, in a "stand-off", A uses a line of reasoning to support his position, but then B reapplies the same reasoning to support a proposition which A cannot accept, forcing A to find another line of reasoning. Such relations organize atomic warrants[8] (as the TRJ Model uses), giving arguments one

[8]Hence "molecules".

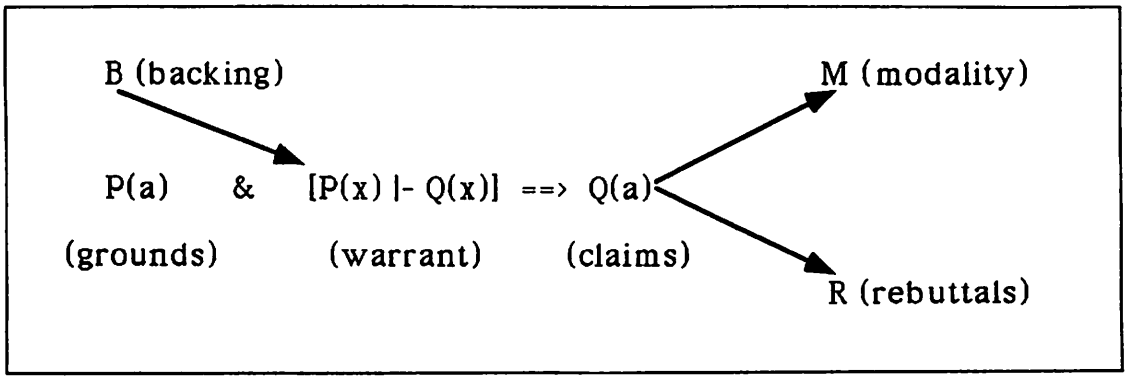


Figure 4.5: The TRJ Model of Argument Structure

more level of structure which, like rhetorical propositions, is vital, yet only implicit in the realization of an argument. Unfortunately the theory as presented is rather simplistic (he only offers two argument molecules, and no computational details of their use), though there are some interesting points made.

Argument Strategies and Tactics

Some analyses of argumentation in legal reasoning have illuminated ways in which arguments (and explanations) may exploit human psychological characteristics. Examples from Dicks (1981) and Rissland (1984b) follow. **Convergence** is the use of several independent ways of justifying the same conclusion. This is logically unnecessary but rhetorically useful in that it increases the apparent validity of the conclusion. Ways of achieving this are **bolstering**, the addition of facts to strengthen a proposition, and **itemization**, listing all relevant reasoning pointing to the same conclusion. Both operate by giving an impression of thorough coverage. A similar strategy is to strengthen the impact of the argument through **repetition** and **restatement** of its propositions. One may manipulate focus of attention in the dialogue by **focusing**, emphasizing certain issues favorable to one's argument, or **obfuscation**, a trick tactic designed to confuse the opponent, by drawing attention away from a weak spot which has just emerged in one's argument. If both sides present equally strong arguments for their own case, the decision is left to the listener's subjectivity. **Refutation**, or presenting an argument against the other's case provides guidance in the desired direction. The opponent's evidence and argument are weakened by **suppression**, suggesting that certain (damaging) data should be prevented or limited, and **mooting**, introducing data or arguments to make another potentially dangerous point or proposition irrelevant. Finally, **vilification** of the opposition questions whether the antithesis is put forth in good faith. These tactics exploit human attentional fallibilities, and rely on various non-deductive ways in which people are influenced towards certain beliefs, e.g. the "amount" of evidence as measured by the number of apparently different ways of supporting a conclusion, the credibility of data, and the merit of the motivations of the participants. The tactics are similar to rhetorical structures, but strike me as more "purely rhetorical": the kinds of effects they are intended to have on the hearer are largely independent of semantic or conceptual considerations.

Cases and Hypotheticals

Many of the argument tactics just discussed may be carried out by citing cases and posing hypotheticals. In case-based domains such as legal reasoning, precedence is all-important, so comparisons between the current situation being argued and other cases form the basis of argumentation. (Ashley, 1987; Rissland & Ashley, 1986) Cases and hypotheticals play a role in understanding the rhetorical structure of argumentation which is similar to that of examples in the epistemology of mathematics (sections 4.1.1 and 4.1.2). The dialectic between principles of fairness, judgments of similarity, legal generalizations or "rules", and the cases themselves drives the development of legal thought (Levi, 1948; Rissland, 1984a). By posing cases and hypotheticals, one can tease apart a situation and motivate new legal concepts and generalizations, much as mathematical examples structure the epistemology of that domain.

4.2.3 Plausible Explanation and Representativeness

The rhetorical techniques discussed above are clearly acceptable in applications such as legal reasoning, where the primary goal is to convince. Are these applicable elsewhere, such as to explaining the reasoning of a problem solver? There is little debate that they are useful as an *additional* level at which an explanation is structured to increase its comprehensibility and appeal. However, some workers at the University of Minnesota (Wick & Thompson, 1988, 1989) take the position that even explanation of expert systems can benefit from being *primarily* concerned with rhetorical techniques for explanation. They point out that human experts often give "stories" which aren't really how they reason, yet are plausible justifications to the hearer. Many users may not understand explanations based on the reasoning of the system, either because they don't understand the domain or because they don't understand the model of computation. I question whether it would be ethical to leave such a user responsible for deciding how to apply the recommendations of a knowledge-based system, unless the machine was capable of teaching the prerequisite domain knowledge and abstracting away from the computations in its explanations. An additional argument for "plausible explanation" is that an attempt to independently reconstruct a conclusion can serve as a check on its validity. This would hold only if the techniques used were ones concerned with validity, and not just apparent appeal.

This is clearly a coupling issue. Given the explanation will be the means by which a problem solver's credibility is judged, how representative should the explanation be of the knowledge and reasoning actually used? While Paris, Wick & Thompson (1988) agree that the "line of reasoning" should be allowed to differ from the "line of explanation" (section 5.2), Paris and the Explainable Expert Systems group (including Moore and Swartout) disagree with the Minnesota group concerning how the two should be coupled, insisting that explanations at least be based on the same knowledge. My position is similar: there is a burden on the explainer to attempt to guarantee that the explanation stands and falls with the original problem solving, so that the problem solving may be judged indirectly, using the explanation as its representative. This is analogous to requiring that the associated conditionals of two different proofs (i.e. the problem solving trace and the explanation) be truth-functionally identical. (Operationalizing this requirement is problematic requirement when the primitive terms of the proofs differ, which is likely to be the case when multiple perspectives are used.) This position implies that the rhetorical techniques just discussed are an additional dimension of explanation, not the sole basis for their structure.

Further work is needed to determine the most fruitful way to define the distinction between epistemological, rhetorical, and other levels of analysis of an explanation. I maintain that theoretical distinctions used to describe explanatory structure should follow distinctions between different sources of constraints on explanation and the knowledge structures on which they operate in an explanation generation process. On this basis, I have argued that rhetorical relations used to describe explanatory text are too bound to surface distinctions to be useful in expressing the final results of a theoretical analysis, though they are useful in the first steps of an analysis for identifying functional roles which must be accounted for. The field is ripe for synthesis of a new vocabulary based on epistemological distinctions, to describe the "deep" structure of explanations and integrate existing research on this structure.

Chapter 5

Generating Explanations

A computational theory of explanation cannot end with an analysis of the structure of explanation, nor even with articulation of the principles behind that structure. It is necessary to specify how such principles are actually used to generate explanations. In this chapter, I discuss computational frameworks for expressing and implementing a theory of how content is chosen, structured, and annotated, resulting in a specification for an explanation. I start by describing and examining the relative merits of two approaches to generating descriptive explanations. One uses schemata which describe the rhetorical structure of typical texts in some abstract form, to be instantiated by replacing the abstractions with the content they select. Another approach exploits the structure of a knowledge-base: the explanatory theory takes the form of procedures for traversing conceptual relations. While efficient for generating isolated texts, both approaches provide poor representations of a theory of explanation, and fail to address certain aspects of interactive dialogues. This motivates current research on planning explanations, which makes the goals and assumptions used to generate the explanation explicit, enabling replanning within a dynamic context.

5.1 Schematic and Procedural Discourse Strategies

This section examines a body of research which resulted in generative theories of the structure of descriptive text. Two techniques were explored and ultimately combined. In the first approach, explanations are generated by instantiating schemata describing the typical rhetorical structure of natural explanations. The second emphasizes the structure of the knowledge-base as a source of constraints on explanation, and extracts this structure through traversal procedures. In this respect, the latter approach is similar to early approaches to expert system explanation (section 2.3), which derived text structure from the reasoning of a problem solver.

5.1.1 McKeown's TEXT

The most influential schematic generation work is that of McKeown (1982, 1985). Her TEXT program answered questions about database structure, by describing what information is available in the data base, providing definitions, and making comparisons between database entities. McKeown operated on two theoretical assumptions. First, the way people describe things is independent of

how information about them is stored in memory. This is to be contrasted with Lehnert's claim (page 14), and McKeown's own subsequent discovery (section 4.1.1) that some descriptions follow the structure of a knowledge-base. However, it does enable her second assumption, that people use standard patterns of text organization to achieve different discourse goals, in a manner only partially influenced by content. Her dissertation:

> "... abstracted the communicative roles played by propositions in a wide range of text used for a common discourse purpose [and] showed how such 'rhetorical schemas' could be used in text generation by attaching a 'semantics' to each rhetorical predicate that indicated the kind of knowledge from the knowledge base that could be used to fill that role. A particular rhetorical schema was chosen based on the purpose of the discourse; the semantics were used to decide what information from a pool of possibly relevant knowledge to include in the discourse; a focusing mechanism was used to arbitrate between choices." (from McCoy, 1986)

TEXT generated text in response to a question in two stages: the strategic component determined the content and structure of the explanation, and the tactical component realized the message in English. I only describe the former.

Discourse Schemata. McKeown's (1982) text analysis used a variety of "rhetorical predicates", discussed in section 4.2.1. Figure 5.1 shows one of the resulting schemata, expressed as a grammar over rhetorical predicates. This is the **Identification** schema, used for providing definitions. Note that the schema also contains an Identification predicate. This permits recursive invocation of the schema, when necessary for generating hierarchically organized structure. McKeown also identified an **Attributive** schema which "illustrate[s] a particular point about a concept or object", **Constituency**, for describing an entity in terms of its parts or subclasses, and **Compare and Contrast**. These may also be expanded recursively when they occur as predicates in other schema. Each predicate, whether recursive or not, also has associated with it a definition of which propositions fill the intended rhetorical role. Simplified versions of these schemata were implemented in TEXT as ATNs (figure 5.2). In addition to the usual `Jump`, `Subr`, `Push` and `Pop` arcs, there were `Fill` arcs which represent a schema's predicates. These were called "fill" arcs because a schema was instantiated by "filling" these arcs, when traversed, with content selected by the test associated with the arc's rhetorical predicate.

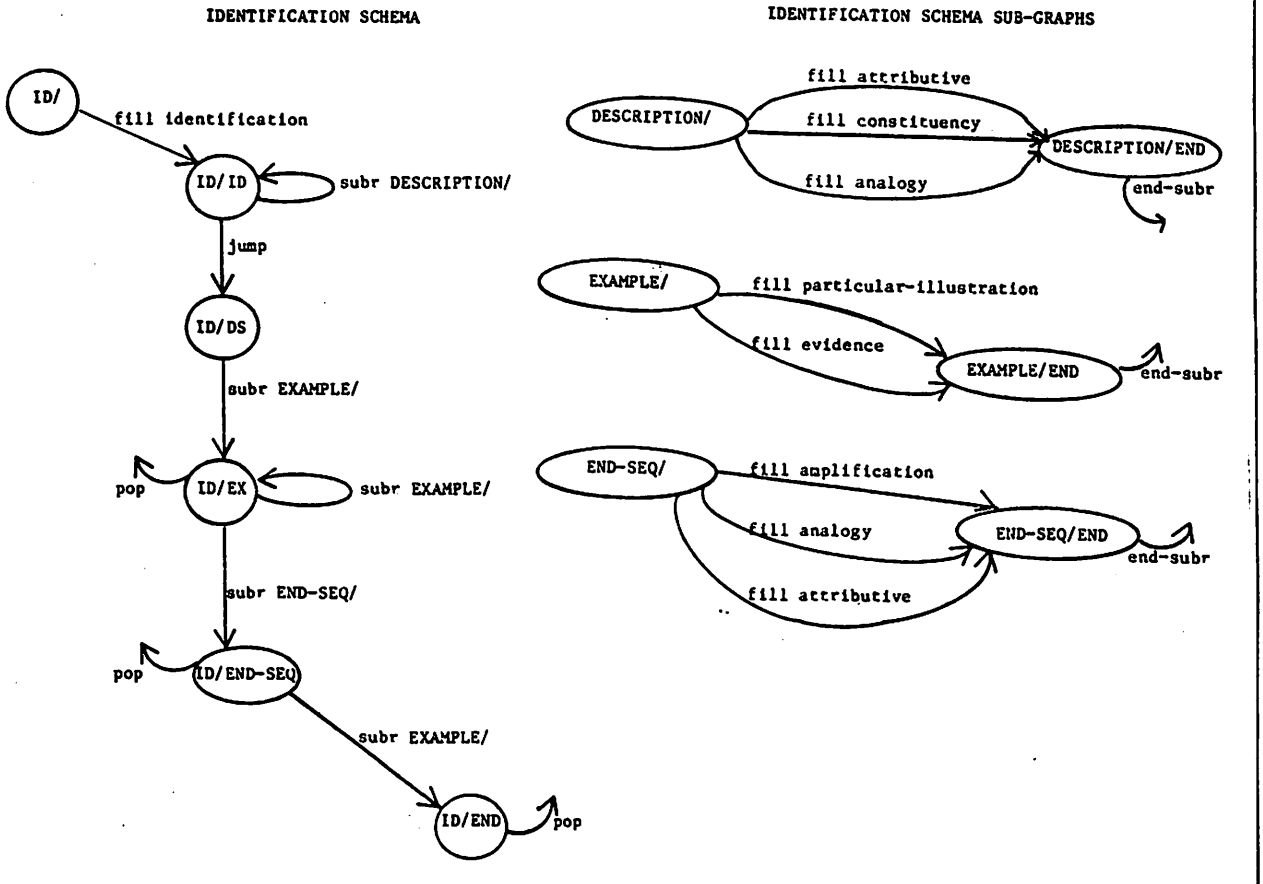
Focus of Attention. Before describing the use of the schemata, I digress to describe mechanisms for focus of attention which are also used to control selection of content. The **relevant knowledge pool** is a means of implementing Grosz's notion of global focus (section 2.2.1). McKeown decided to err on the side of including as much information as might be relevant, assuming a naive user of the system. Selection is sensitive to the question type. For an information or definition request, all semantic network links out of the questioned object are included. Also, all of its siblings and descendants in the hierarchy are included along with all their links, in case they are needed for analogies or constituency descriptions, respectively. Comparison questions result in a selection process which is sensitive to the "closeness" of the objects in the hierarchy — the style of selection is similar. McKeown does not attempt to deal with choosing between multiple perspectives, or simplifying the knowledge pool for the sake of a user who cannot handle great detail.

Local focus is based on an adaptation of Sidner's (1979) work (section 2.2.1) to generation. McKeown uses the following **preference ordering on focus shifts** in a stack discipline:

```
Identification:
   Identification
   {Analogy | Constituency | Attributive | Renaming}*
   Particular-illustration | Evidence+
   {Amplification | Analogy | Attributive}
   {Particular-illustration | Evidence}
```

Key: {} — optional; | — disjunction; * and + — Kleene star and plus.

Figure 5.1: The Identification Schema



From McKeown (1982), p. 73.

Figure 5.2: ATN Representation of Identification Schema

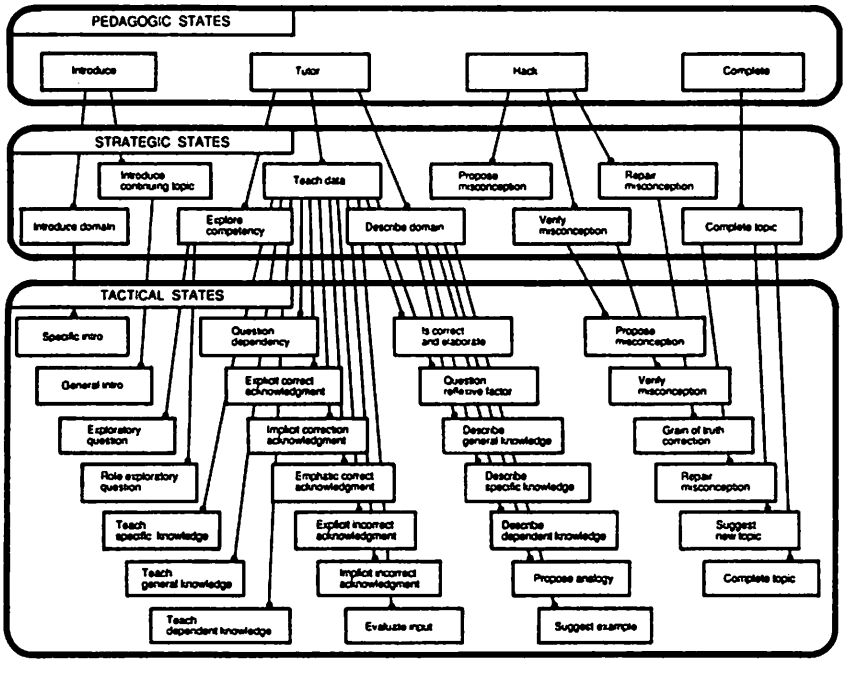
1. **Push.** Prefer to shift focus to a new subtopic just introduced. This is guaranteed to terminate because the relevant knowledge pool is finite, and eventually will be exhausted. (Problems could arise with this heuristic given a large and detailed knowledge-base.)
2. **Maintain.** Otherwise maintain the current focus. Since a "pop" closes the current focus, and there is no provision for returning to its topic, maintaining focus is preferred so propositions in the relevant knowledge pool are used up before closing the topic.
3. **Pop.** When the relevant knowledge of the current focus is exhausted, return to the previous focus.

If a conflict remained after application of the above rules (in priority order), then McKeown's algorithm selects the proposition with the greatest number of implicit links to previous propositions, where an implicit link is (as far as I can tell) a path between two topics in the semantic network. No attempt is made to filter or weight these links according to their relevance to the global focus. In contrast with this strict stack discipline, recall how BLAH (Weiner, 1980; section 2.3.2) preferred to finish the main point first, without cluttering the explanation with detailed subproofs (figure 2.5).

Using the Schemata. Schema selection is accomplished by associating schemata with question types in a manner sensitive to what information is available. If the user wants to know what information is available about an object, either the Attributive or the Constituency schema is used. The latter is selected when the object is above a predetermined level in the class hierarchy, since it will have more constituents than specific attributes. Similarly, requests for definitions are answered with the Identification or Constituency schema. The Compare and Contrast schema is used to describe the difference between two objects. The hierarchical-level heuristic seems reasonable for an application which is only meant to tell the user what a database system contains. However, Constituency would not be selected for objects low in the hierarchy which have many physical parts. McKeown is not addressing the structure of explanations which describe the parts of objects themselves, presumably since this is the job of the database query facility.

The selected schema is filled by traversing the ATN, using the predicate semantics to select propositions from the relevant knowledge pool when an arc is traversed. Each proposition used is removed from the pool, to avoid repeating it. Where there is a choice of which arc to take, or which matching predicate to use, the focus constraints discussed above are used to select the most appropriate predicate (and hence arc). Thus, in her approach explanatory structure comes from two sources: common rhetorical patterns (given without theoretical justification), and explicit focus constraints.

McKeown's work influenced many others to use discourse schema (Cawsey, 1988; Maybury, 1988; McCoy, 1986; Paris, 1987; Roth, Mattis, & Mesnard 1988; Winkels, Brueker, & Sandberg, 1988; see also next section). I have mentioned a few deficiencies in McKeown's theory; she also acknowledges that she does not use discourse history or a user model. Some attempts to remedy this are described in the next few subsections. While I will conclude that schemata are limited from a theoretical point of view (section 5.1.5), McKeown's work was significant for a distinct reason: her "rhetorical predicates" helped raise the level of abstraction at which explanation theories are expressed. However, as noted in section 4.2.1, these predicates mixed conceptually distinct considerations.



From Woolf & McDonald (1984).

Figure 5.3: Discourse Management Network Hierarchy

5.1.2 Variations on Schemata

Schemata as conceived by McKeown fail to capture the rich structure of explanation from different levels of analysis. This is partly due to the nature of her rhetorical predicates, and partly because schemata compile into one data structure a variety of considerations. Also, McKeown's schemata were not interpreted in a manner sensitive to the user and dialogue context. In this section I review some work which attempted to address these problems while retaining the schematic approach.

Meno-Tutor. Woolf & McDonald (1984) augmented the ATN formalism to achieve sensitivity to dialogue and user, and better express the structure of tutorial dialogue. The usual states of an ATN are called **tactical states**, and are used to represent tutorial "moves" (descriptive, questioning, or acknowledgment acts). These appear to mix speech acts with abstract content specification. Collections of tactical states are abstracted into pedagogical and strategic states by a **discourse management network** (DMN, figure 5.3). **Pedagogical states** represent "specific tutoring approaches", which appear to be phases of a tutoring section, while **strategic states** represent "the approach to be used", differing in style such as the relative use of descriptions and questions. The paper lacks a principled definition of pedagogical and strategic states, and gives examples which make the two appear similar. While the instances of these states were empirically derived from tutorial dialogues, it is unclear whether the pedagogical vs. strategic distinction itself provides theoretically useful levels at which to describe the structure of explanatory dialogue, or whether the

Misconception: X is-a Y
Response:
1. X is-NOT-a Y
2. X is-a SuperType(Y)
3. X is like Y because both share (intersection (attributes-of X) (attributes-of Y))
4. BUT X has (set-difference (attributes-of X) (attributes-of Y))
5. WHILE Y has (set-difference (attributes-of Y) (attributes-of X))

Figure 5.4: An Abstract Misconception Response Strategy

```
((deny (classification OBJECT POSITED))
  (state (classification OBJECT REAL))
  (concede (share-attributes OBJECT POSITED ATTRIBUTES1))
  (override (share-attributes ----- POSITED ATTRIBUTES2))
  (override (share-attributes OBJECT REAL ATTRIBUTES3)))
```

Figure 5.5: A Schematic Representation of the Strategy

layered mechanism is to be credited for increased expressive power obtained merely by enabling one to group tactical states as one sees fit.

The DMN is first used as a refinement structure for selecting an entry point into the ATN at the tactical level. Then transitions in the ATN occur as in other systems. However, a set of **tutoring meta-rules** which test the student and discourse models enable the system to jump between ATN states at any level of abstraction in the DMN. This is intended to model the way in which human tutors dynamically change their approach according to the behavior of the student. Tactical schemata represent standard patterns of interaction, while meta-rules represent when to change ones' approach. Woolf & McDonald point out that while the preemption paths taken by meta-rules could in principle be encoded into the ATNs, the point of this mixed representation was to allow experimentation with tutorial strategies in the meta-rules, which are expressed at whatever level of abstraction in the DMN appears appropriate.

Responding to Misconceptions. McCoy (1985, 1986) uses discourse schemata to specify the content and structure of corrective responses to misconceptions (defined as discrepancies between the system's knowledge-base and the apparent beliefs of the user). Her paper focuses on misclassifications. McCoy's schemata separate specifications of the "rhetorical force" of the explanation from content selection, and she includes mechanisms for sensitivity to discourse context and a user model.

According to McCoy, a misconception (if not ignored) can be responded to by **denial** of incorrect information; **statement** of correct information; and/or **justification** for the denial/correction given. The "Gross level strategy" chooses some subset of these to do, based on the discourse goals (eg. you may ignore a misconception that is not relevant to the goal; or omit justification when correcting a misconception of minor importance). Then, the "Strategy Elaborator" tries to fulfill the request, using "fine level" strategies.

Abstracting from some example dialogues, McCoy gives the strategy in figure 5.4. While this specifies the content of the response, she notes that it does not capture the communicative role played

by each part of the response. This information would aid in appropriate text generation. Hence, McCoy annotates each of the propositions to be expressed with "rhetorical force predicates". The resulting schematic representation of the strategy is shown in figure 5.5. This schema is chosen when there is evidence that the user believes that the misclassified OBJECT (X of figure 5.4) is similar to the POSITED superordinate (Y). It directs the text generator to deny that the object is a member of the class posited by the user, state the real classification, concede that the object shares attributes[1] with the posited class, but override this concession by pointing out attributes exclusive to each. The gross-level strategy may elect to omit some of these communications, and the text generator may elect to reorder them. Due to this flexibility, the schemata actually provide almost no information about explanatory structure, let alone a theoretical basis for deriving it. They only specify what the content could be, and its communicative role in terms of rhetorical force.

Note that the force predicates of figure 5.5 are actually illocutionary speech acts (section 2.2.2). In spite of this, McCoy says these predicates were derived from the work of McKeown. She believes her work differs by abstracting content, while McKeown only abstracted communicative roles. I believe it is more accurate to say that McKeown ***abstracted*** both roles and content, but failed to ***distinguish*** them in her rhetorical predicates. Rhetorical predicates are an insufficient basis for expressing a theory of explanation, and are in need of further analysis to break them into the levels of explanatory structure they summarize. McCoy has separated illocutionary roles from the content specification, and hence has taken a first step in this direction.

A third level of explanatory structure has to do with its ties to discourse context. McCoy argues that attributes which have been highlighted by the discourse context should play a stronger role in the similarity comparisons used in schemata such as that in figure 5.5. This is achieved by object perspective (section 3.2.2), used to weight the attributes in the set operations of figure 5.4.

5.1.3 Procedural Strategies for Object Description

In section 4.1.1, I introduced the idea that explanations should be ordered to follow the structure of knowledge. This idea had already been illustrated earlier in the paper: in early ITS research, explanations were generated by traversing semantic networks (section 2.1.1); and most early approaches to explaining problem solving were also expressed as procedures for traversing and reorganizing the problem solver's trace (section 2.3). It is natural to represent explanatory strategies which exploit the structure of a knowledge base as traversal algorithms. Here I return to more recent research by McKeown and one of her colleagues to illustrate this approach, and discuss the utility of procedural representations of a theory of explanation.

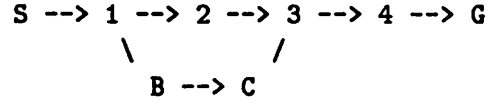
In attempting to characterize descriptions of physical objects, Paris and McKeown (1986) concluded that they had to abandon **declarative strategies** such as the schemata used in TEXT. Object descriptions are sensitive to the structure of the objects (and hence the content of the knowledge-base), rendering imposition of rhetorical structure by schemata inappropriate. Instead, they used a **process strategy** which directs how to traverse the knowledge structures describing the object.

The knowledge-base is assumed to contain detailed spatial, functional, causal, and attributive

[1]The source of ATTRIBUTES1, etc. is not explained, but presumably these are computed by the set operations of figure 5.4, followed by object perspective filtering (section 3.2.2).

information about devices obtained[2] from patent applications. The process strategy generates a description of how a device works, so is concerned with links between events. The first step is to categorize the causal, temporal, and "analogical" links in the knowledge-base, and rank them by order of importance to a process description. The **main path**, connecting an initial state or event to a goal state, is then found by search over these links, with conflict resolution and backtracking ordered according to the link ranking. Side paths are also kept track of, including lower ranked ones that re-intersect the main path.

The process description follows the main path. However, the algorithm must decide whether to include side paths. They may illustrate the explanation, or might reach important side effects relevant to interlocutor's goals, or could later enable states on the main path. Paris & McKeown classify pathway patterns according to abstract, graphical properties. Prescriptions for when and how to include side paths in the description are then given in terms of this classification. For example, one pattern is a long side path (1, B, C, 3) that reintersects the main path (S, ... G):



If the path is important, eg. the event at the intersection 3 is *enabled* by the side path from C, then the side path is traversed using a focus shift, followed by return to the main path. An isolated side link is used only if it is an essential side effect or an analogy providing a clearer explanation. If there are many short side links, they are either ignored or collected to be described after the main path. While Paris & McKeown have abandoned schematic discourse strategies for object description, at the implementation level they retain the use of ATNs for strategy representation. Use of the same declarative formalism facilitates employing both strategies in the same description (see section 5.1.4).

This work provides some techniques which would be of value to someone needing to implement a simple yet potentially general way of generating process descriptions from knowledge bases utilizing causal and temporal relations. However, I believe that these search and pruning algorithms are not an appropriate way to express a theory of explanation. They suffer from the same deficiencies as EXPOUND and BLAH, and indeed McKeown's declarative schemata, in being the result of rather than the expression of such a theory. The focus is on how to generate an explanation, without saying what explanatory goals are being met and what the explanation assumes about the interlocutor. Issues such as which knowledge structures to use and how to linearize them into an explanation have already been decided and designed into the procedure. This results in a potentially efficient mechanism, but works only because the explainer's context is restricted to object descriptions using a single-perspective knowledge base which leaves few choices to be made. In the next section I examine Paris' attempts to improve flexibility via a meta-strategy which mixes this process strategy with McKeown's schemata.

5.1.4 Mixing Strategies

Based on a study of encyclopedias intended for different audiences, Paris (1985) showed that persons unfamiliar with a domain need process descriptions, saying how a device performs its function,

[2]From an automatic parser they had not yet built. It was pointed out to me that their work assumes the parser was able to generate a complete description of the object from the applications. If this is violated, crucial causal paths may be missing or incomplete, and hence mistreated by the algorithm about to be described.

while experts prefer structural information about the components of a device and their attributes. (As illustrated in section 3.2.2, the ability to infer process from structure is an important aspect of expertise.) Since both the process strategy of Paris & McKeown (1986) and the constituency strategy of McKeown (1982, 1985) are represented in the same ATN formalism, they can be chosen between and even mixed for an intermediate user. Paris (1987) reports on how to do this.

Paris assumes that a user model distinguishes the user's level of expertise at various granularities, e.g. with respect to sub-areas of the knowledge-base and to individual objects, and indicates whether the user is familiar with basic concepts which may be included in a process description. At the coarsest granularity, the starting schema is chosen in a straightforward manner, essentially by using constituency for experts and a process trace for novices. Once a schema is chosen and in use, there are several situations in which strategies may be switched at finer granularities:

- Any time a new object is introduced.
- Within the Constituency Schema: After identifying an object as a member of a generic class, a switch to the process strategy for that class is allowed. After mentioning the parts of an object, a functional (instead of structural) description may be used for the parts.
- Within the Process Trace: The constituency schema may be substituted where the process trace specifies that attributes be used. When describing subparts, constituency can be substituted for functional information.

The heuristic for deciding *when* to take these legal switches is the same as for choosing the initial strategy at top level, based on levels of expertise in the user model.

The ability to switch strategies is clearly important. Paris has provided some reasonable heuristics for doing so, a contribution towards understanding the epistemological structure of explanation. Her meta-strategy does rely on a quite detailed user model. Given the intractability of user modeling (Self, 1988), her approach would be helped by heuristics for choosing strategy based on other considerations, relying on assumptions about the user and planning further explanations if needed.

The use of one mechanism, the ATN, for both strategies is good program design, but is not as unifying as it might seem. Examining the mechanisms which are *implemented in* the ATN, Paris still has declarative strategies which force content into hand-coded templates, and procedural strategies which follow links in the knowledge base. Hence the meta-strategy is needed to coordinate these two mechanisms for generating explanations. The meta-level strategy has more fundamental importance in its potential for a unified representation of the basis on which selection and structuring decisions are made. If Paris refined the meta-level strategy to be more flexible and rely less on the user model, it would exert finer control over the explanations. I would expect the descriptive and process strategies to wane in importance, becoming merely primitives for accessing a knowledge base, as more of the reasons for the choices they made implicitly were brought into the explicit meta-level strategy.

5.1.5 Evaluation of Descriptive Approaches

Schematic and procedural discourse strategies are at two extremes with respect to the derivation of the structure of an explanation. One imposes structure (albeit one derived from study of many natural explanations), while the other extracts existing structure. In defense of schemata, they do provide an easy to comprehend answer to the question "what is an explanation", at least descriptively

with respect to certain considerations of content and rhetorical structure. For example, reading off the Identification schema (figure 5.1), one can say that explanations used to define a term typically name the class it belongs to, describe constituents and/or gives attributes of the referent, then give an illustrative example. A procedural specification for traversing a knowledge-base doesn't wear its sense of explanation on its sleeve, though it does tell us that the "natural" structure of knowledge constrains the structure of an explanation.

These two approaches are conveniently described together because the same researchers developed and eventually integrated b. th of them. Also, despite their apparent differences, they are similar in a more fundamental way. Schemata don't justify their structure; in particular they don't give the goals and assumptions behind their structure. Traversal procedures don't reason about what structure to exploit for a given explanation, and why. In this sense, though able to generate text, declarative schemata and process strategies both remain descriptive at a theoretical level. A more complete theory would achieve the same effect as the schemata or traversal procedures by generating knowledge retrieval goals from more basic rhetorical and epistemological principles of communication. If efficiency were an issue, such a theory could be used to generate and hence justify schemata and traversal procedures.

This is not to detract from the contribution of the work just reviewed. The deficiencies of descriptive techniques may be tolerated in a research program which is at the point where it is necessary to be descriptive just to get a handle on the problem. (It is not surprising that the researchers in section 5.1 derived their schemata empirically.) Subsequent research will gradually "decompile" the schema or procedures into deeper principles from which the surface forms may be generated.

5.2 Planning Approaches to Explanation

In this report, I have illustrated how various kinds of knowledge are needed for complete explanation facilities. A good explainer should be able to decide when to include such knowledge in such a manner that facilitates the interlocutor's understanding, and cannot simply take the surface structure of reasoning as the sole guide to explanatory content and structure (sections 3.1 through 4.1). Furthermore, linguistic and rhetorical constraints (sections 2.2 and 4.2) should be taken into account. The coordination of all these considerations won't be solved by more ways of traversing and transforming an application problem solver's trace, nor by fixed schemata. Explanation researchers[3] generally agree that explanation should be treated as problem solving in its own right (Paris, Wick, & Thompson, 1988). Several researchers are currently examining the use of planning formalisms, including Appelt (1985), Bienkowski (1986), Cawsey (1988), Hovy (1988), and Moore & Swartout (1988, 1989), as well as myself. This section will clarify why planning has important advantages for generating the content and structure of explanations, and provides a better context within which to develop a theory of knowledge communication.

5.2.1 Planning Communicative Acts

An influential study of planning communicative acts is due to Appelt (1985). His theory, embodied in a system called KAMP, focuses on the planning of English **referring expressions**, intended to

[3] In particular, participants of the AAAI-88 workshop on explanation.

activate concepts in the hearer's mind. The domain is one in which a computer agent, who can perform no physical actions, plans utterances to get a human to disassemble a pump, for example: "Remove the pump with the wrench in the tool box". Appelt's theory emphasizes reasoning about belief, mutual belief, intention, and recognition of intention as the basis for choosing the illocutionary acts needed to get another agent to do something, and identifying the propositional acts needed to provide the agent with the necessary information to carry out the desired act. The planning of illocutionary and propositional acts occurs independently of planning sentences and noun phrases (i.e., there is not a one to one correspondence), and these acts are realized in surface speech acts which satisfy multiple communicative goals. Appelt is mostly concerned with linguistic issues which I am ignoring in this review, but his work has influenced those planning explanatory content at a conceptual level as well.

Appelt's approach is based on Cohen & Perrault's (1979) characterization of speech acts as operators in a planning system. Planning occurs at four levels of abstraction: **illocutionary acts**, i.e. *inform* and *request*, whereby intentions are communicated; **surface speech acts**, i.e. *command*, *ask*, and *assert*, which are abstract representations of physical utterances; **concept activations**, the referring expressions which ensure that concepts used to construct propositions are in the current focus of attention; and **utterance acts**, the actual language planned. Two versions of the actions available to the planner are used: a complete axiomatization of Appelt's theory based on possible worlds semantics, and STRIPS-like[4] action summaries, which make heuristic simplifications necessary to reduce the complexity of planning. Initially, the hierarchical planner expands communicative goals in a top-down, context-free manner, using the action summaries. After each expansion, critics examine the plan, identifying and fixing context sensitive interactions. Once the plan is complete at a given level of abstraction, the axiomatic theory is applied for a detailed proof of correctness, using worlds associated with nodes in the procedural network. If the proof failed, replanning would have to occur at a more detailed level.

Appelt elegantly combines a variety of existing techniques and theory; e.g. his planner uses mechanisms such as procedural networks, action summaries, and critics; and his theory integrates possible worlds semantics, reasoning about action and mutual beliefs, speech act theory, and other linguistic concerns. His paper illustrates how a planning system provides a uniform computational framework within which to examine the interaction of a variety of constraints on the generation of communicative acts. Appelt believes that content planning and linguistic realization should be closely coupled, threatening the idea that explanation research can study the structure of explanation at conceptual and epistemological levels without bothering with surface-linguistic concerns. However, appropriate use of levels of abstraction in an extensible planning system may enable this separation, for example by taking speech acts as the primitive actions and leaving open the possibility that later addition of expansion into utterance acts will generate further constraints on planning at the higher levels. A limitation of Appelt's theory is the assumption that the model of interlocutor belief is correct, and that the utterance will be understood. It seems unwise to expend much effort for a proof of "correctness" when no guarantee can be made, and the interlocutor is available for feedback.

5.2.2 Interactive Explanation

Until recently, planning has been used only to generate isolated utterances, and attempts to generate the perfect explanation the first time. This approach to explanation assumes and relies on a better

[4]STRIPS was a planner which represented plan operators in terms of their preconditions and add and delete lists, which specified what formulas to add to or remove from a data base representing the state of the world.

user model than one can expect. A more realistic approach is to use whatever is available in the user model, but be able to operate without this information. Feedback is a less costly source of information about the interlocutor, which can only be exploited by a model of explanation which takes interactive dialogue context into account. Moore & Swartout (1988) suggest that the following capabilities are needed to deal with interactive explanation:

- Monitor the effects of utterances on the interlocutor.
- Recover if feedback indicates dis-satisfaction with response.
- Answer follow-up questions in the dialogue context, not as an isolated question.
- Offer further explanations even when a follow-up question is not clearly formulated.
- Try different response strategies or change perspectives as needed.

As discussed in section 5.1.5, schematic and procedural approaches generate explanatory acts without saying why they are appropriate. There is no provision for recovery from failure to explain, since the system doesn't know what goal it was trying to achieve nor what assumptions might be responsible for the failure. Even if the schemata or procedures were annotated with such information, additional mechanisms would be needed to use it. In contrast, planning permits one to make explicit the ways in which various heuristics, constraints, and assumptions are combined to achieve explanatory goals. This enables the planner to redo what it has done, if necessary, without additional mechanisms, and encourages inclusion of these considerations in one's theory. The next section examines Moore & Swartout's version of explanation planning, which begins to realize these advantages.

5.2.3 Reactive Planning

Moore & Swartout (1988, 1989) are extending the single-utterance techniques of Appelt (1985) and others to handle longer explanations in context of an ongoing dialogue. In their approach, explanation planning uses a library of rule-like explanation operators, and selects rhetorical structures and speech acts as well as the content necessary to achieve its explanatory goals. The plan records assumptions made about the interlocutor in choosing an operator, and notes alternate operators available but not used. These annotations are used when replanning.

The Plan Language. A **plan operator** (exemplified in figure 5.6) consists of these components:

Effect: the goal (discourse goal or rhetorical relation) the operator can achieve.

Constraint List: preconditions on facts in the knowledge-base or model of the interlocutor's beliefs.

Method: a sequence of steps which achieve the effects of the operator. As in RST (section 4.2.1), all methods have a nucleus, and some may have optional or required satellites.

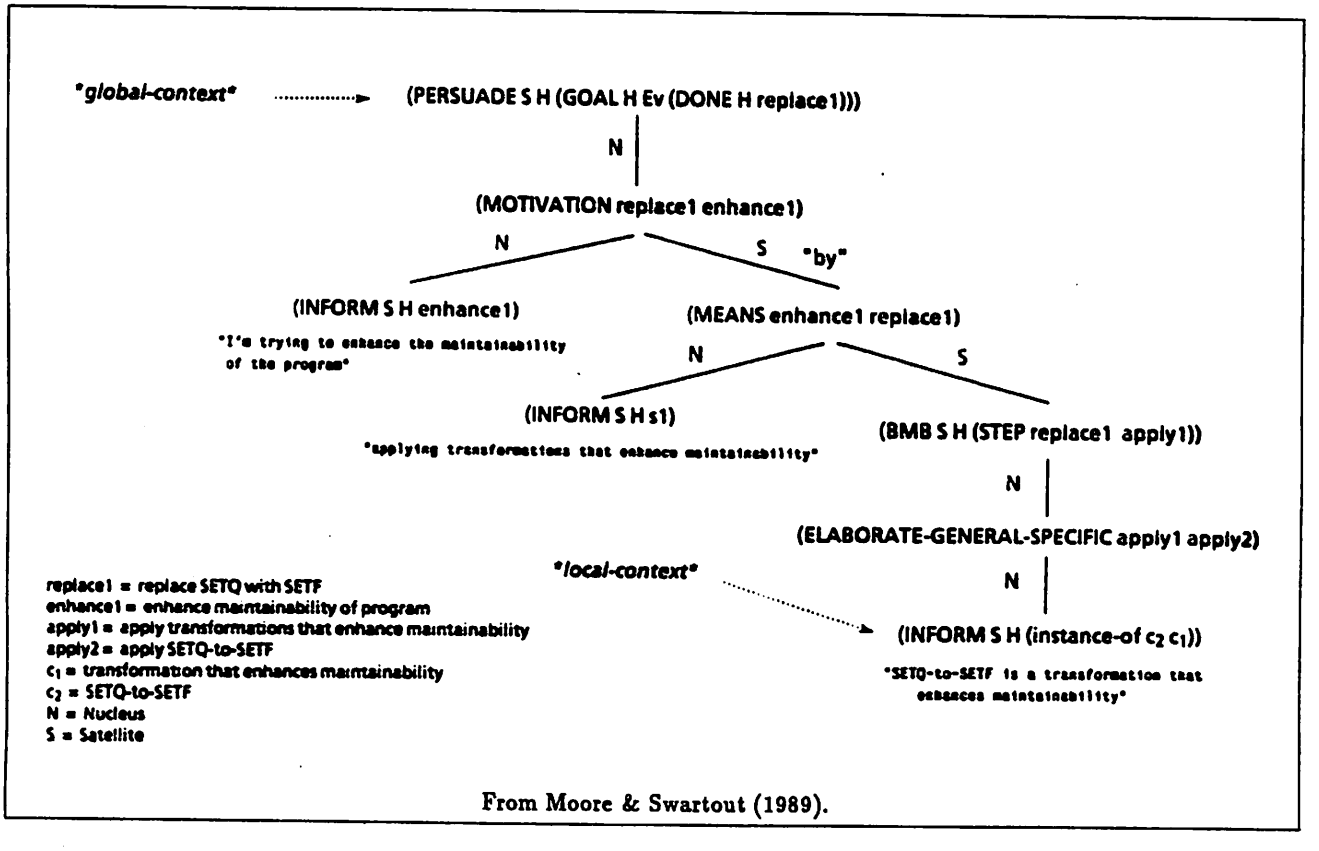
Speech acts are the primitive actions. (INFORM and RECOMMEND are the only two speech acts implemented at this writing.) While Moore & Swartout distinguish speech acts, discourse goals, and rhetorical relations, the latter two are both represented as planning operators without a clear statement of the difference between them. Discourse goals are assertions which the system wants the interlocutor to believe, while rhetorical relations are embodied in the way the plan operators break discourse goals into subgoals and speech acts (relying on a variety of implicit heuristics and constraints in doing so).

EFFECT: (PERSUADE S H (GOAL H Ev(DONE H ?act)))
CONSTRAINTS: (AND (GOAL S ?goal)
(STEP ?act ?goal)
(BMB S H (GOAL H ?goal)))
NUCLEUS: (FORALL ?goal (MOTIVATION ?act ?goal))

EFFECT: (MOTIVATION ?act ?goal)
CONSTRAINTS: (AND (GOAL S ?goal)
(STEP ?act ?goal))
NUCLEUS: (INFORM S H ?goal)
SATELLITES: (((MEANS ?goal ?act)))

From Moore & Swartout (1989).

Figure 5.6: Plan Operators



From Moore & Swartout (1989).

Figure 5.7: Completed Plan

Operation of the Planner. Planning is directed towards discourse goals derived from the reasoning of an expert system, or from the interlocutor via a query analyzer. The planner operates as follows:

1. A discourse goal is posted.
2. Candidate operators are identified:
   (a) Operators whose effect matches the goal are identified.
   (b) Candidates are those whose constraints are satisfied or assumed. Constraints on the contents of the expert system's knowledge-base must be satisfied, but those on the user model may be marked as assumptions, if not directly satisfied.
3. An operator is selected, based on the user model and dialogue history. The **selection heuristics** prefer operators with the following characteristics:
   - More specific plan operators.
   - Minimal assumptions about the interlocutor's beliefs.
   - Use of a concept the interlocutor knows.
   - Use of a concept mentioned in the dialogue history.

   Unchosen operators are recorded as alternatives.
4. The nucleus and required satellites are posted as subgoals to be expanded, unless they are executable speech acts. (Some mutual belief goals may be satisfied by the user model without further planning.) Expansion of optional satellites depends on the user and dialogue models, e.g. they are not expanded unless the interlocutor is a novice, or frequently asks "why?".
5. Planning is complete when all posted goals are expanded.

The resulting plan is recorded in the dialogue history and passed to a text generator. Figure 5.7 gives an example of a plan, in the Program Enhancement Advisor, a system for helping people improve the style of their LISP programs.

Query Handling and Recovery. Moore & Swartout keep track of the system's discourse goals to disambiguate references (e.g., when the user says "Why?"), or to identify what the interlocutor doesn't understand if the question is vaguely articulated ("Huh?"). One heuristic for disambiguating follow-up questions is to prefer continuing on the current topic over returning to a previous topic (as in McKeown's TEXT). Another is "tell the user what he's least likely to know", taking advantage of what may happen to be in the user model. If the local focus of attention contains something the user already knows, the system searches up the focus stack (derived from the plan) to find the first information which may not be known by the user. The prioritized[5] **recovery heuristics** for recovering from failure to explain are:

1. If any assumptions were made in planning the last explanation, post goals to make these true (i.e., plan to tell the user what the system previously assumed he/she knew).
2. If other plan operators exist for achieving the discourse goal, but were not tried, replan using one of these operators. If the discourse goal is to define a concept, additional selection heuristics are used: prefer to given an example; if that fails, use an analogy if one is available.

[5]Priority order based on personal communication from J. Moore.

3. Expand any unexpanded optional satellites in the previous plan operator.

For example, in figure 5.7, if the user has asked "why" at `*local-context*`, the planner must determine whether the user is asking why `setq-to-setf` enhances maintainability, or why we are trying to enhance maintainability by applying transformations that do so, or why the program is trying to enhance maintainability at all. If the user model doesn't show that the interlocutor knows why `setq-to-setf` enhances maintainability, and this was assumed while generating the plan, the planner makes good on this assumption by planning to tell the user why this replacement has the desired effect. Suppose this leads to the user asking what a "generalized variable" is, and the user responds with "huh?" after being provided with a definition. An example would then be given, based on the second recovery heuristic.

Features relegated to future work include a richer dialogue history; determining when things are "forgotten" from the dialogue history; finer grained interleaving of text planning with execution so that execution monitoring can occur at this granularity; considering pragmatic goals; and mixed media planning. There is probably some interesting epistemological theory implicit in their heuristics waiting to be articulated. The effort which Moore & Swartout put into disambiguating query references is necessary for modeling human speech, but in a computer interface this could be replaced with a simpler method of diexis such as mouse selection of the intended referent.

5.2.4 Schemata as Plan Operators

Hovy (1988a,b) also uses RST in his **Structurer**. The Structurer is given both a top level communicative goal and the content of the explanation (a pool of propositions), and structures the content into a coherent text plan. The plan is a tree of rhetorical structures, matching the goal to as much of the content as possible. Hence the Structurer is less ambitious than Moore & Swartout's planner, which plans content as well as structure. Hovy's "RST relation/plans" (exemplified in figure 5.8) are more complex than Moore & Swartout's plan operators, and may be read as mini-schemata, demonstrating that there is no clear line between schemata which invoke other schemata and planning by composing plan operators. Hovy suggests that his RST operators may be used in either a "top-down" or "open-ended" manner, depending on whether "growth points" (the satellite subgoals) are seen as injunctions (giving rise to schemata) or as suggestions for inclusion of further material, to be decided based on other criteria applied by a planner. Clearly, planning is potentially more flexible. However, the work of Moore & Swartout shows that the advantage of plan operators over schemata is not limited to bringing flexibility down to a finer granularity. In deciding whether to include optional material, a record of the goals and assumptions relied on has utility in an interactive context.

Hovy (1988b) lists some criteria for inclusion or exclusion of optional material, which I reproduce in figure 5.9 with the addition of my own categorization of his criteria according to the level at which they are operating (compare to the material of section 6.1). No explanation theory to date has handled these considerations in an explicit and principled manner, though the planning approaches just described are making first steps.

The above discussion should have made it clear that planning can do what a schematic approach can, since a planner can essentially piece together different schemata out of its plan operators. A

```
Name: SEQUENCE

Results:
  ((BMB SPEAKER HEARER (SEQUENCE-OF ?PART ?NEXT)))

Nucleus requirements/subgoals:
  ((AND (BMB SPEAKER HEARER (MAINTOPIC ?PART))
        (BMB SPEAKER HEARER (NEXT-ACTION ?PART ?NEXT))))

Nucleus growth points:
  ((BMB SPEAKER HEARER (CIRCUMSTANCE-OF ?PART ?CIR))
   (BMB SPEAKER HEARER (ATTRIBUTE-OF ?PART ?VAL))
   (BMB SPEAKER HEARER (PURPOSE-OF ?PART ?PURP))
   (BMB SPEAKER HEARER (DETAILS-OF ?PART ?DETS)))

Satellite requirements/subgoals:
  ((BMB SPEAKER HEARER (MAINTOPIC ?NEXT)))

Satellite growth points:
  ((BMB SPEAKER HEARER (ATTRIBUTE-OF ?NEXT ?VAL))
   (BMB SPEAKER HEARER (DETAILS-OF ?NEXT ?DETS))
   (BMB SPEAKER HEARER (SEQUENCE-OF ?NEXT ?FOLL))
   (BMB SPEAKER HEARER (PURPOSE-OF ?NEXT ?PURP)))

Order: (NUCLEUS SATELLITE)
Relation-phrases: ("" "then" "next")
Activation-question:
  "Could ~A be presented as start-point, mid-point, or
   end-point of some succession of items along some
   dimension? -- that is, should the hearer know that
   ~A is part of a sequence?"
```

From Hovy (1988b).

Figure 5.8: SEQUENCE Plan Operator for Hovy's Structurer

Level	Inclusion Reasons	Exclusion Reasons
Information	Explicit goal to communicate extra material	Lack of known detail
	Hearer asks for more information	Untrustworthyness of material
Epistemological	Hearer cannot understand without the extra material	
Stylistic	Balance and parallelism of text	Unpleasant or undesirable effect of material
Pragmatic		Lack of time or space

Figure 5.9: Hovy's Inclusion and Exclusion Criteria

planner can also exploit the structure of knowledge, simply by using operators which follow this structure while constructing a plan. For example, a plan operator which is only enabled when there is a goal to describe a process from a causal perspective could construct a sequence of utterance acts following a causal chain describing that process. Variables in the operator's condition could bind to the incomplete causal description in the explanation plan and to a matching point in the knowledge base, to identify the next causal step and where to add it to the explanation. It not necessary to go to other mechanisms such as schemata and procedural strategies for generating explanations — one can treat explanation in a uniform planning framework. This provides for a clearer expression of one's theory, and allows an examination of the interaction of content selection and ordering activities which previously were isolated from each other in distinct mechanisms.

Computational efficiency and the necessity of writing an adequate set of plan operators are concerns which may make or break the planning approach. However, my own opinion is that planning is the most promising mechanism with which to study explanation. A variety of previous work is being integrated in these projects, and explicit reasoning about explanation goals, constraints, and assumptions is a framework within which much more could be done. It will be interesting to see how conceptual, epistemological, rhetorical, and linguistic considerations interact in an environment which reasons explicitly at all these levels to produce an explanation. While restricted applications may need to get by with compiled methods such as schemata and traversal algorithms for the sake of efficiency, our theoretical understanding is better advanced by approaches which reason from first principles and make the distinctions and interactions between these levels explicit. In the next chapter, I try to bring together and summarize the levels at which an explanation can be analyzed and planned.

Chapter 6

Issues for Explanation Research

The paper so far has emphasized the analysis of previous explanation research. In contrast, this chapter is a synthetic offering of my own observations and conclusions. It is intended to summarize and organize what the literature has shown about the central issues outlined in section 1.2, and to suggest further research directions.

6.1 A Framework for Constraints on Explanation

In Chapter 1, I deferred on defining what an explanation is, but suggested that a useful handle on defining a good explanation may be to characterize the heuristics and constraints which are applied when generating one. Throughout the review, I have presented such criteria which occur at different "levels" of analysis. Here I sketch a framework within which to place these various heuristics and constraints, characterizing them in terms of where they come from and what part of explanation planning they bear on. I introduce this analysis with a descriptive summary of the attributes of a good explanation, written to illustrate the framework to be discussed. I then examine parts of this description in terms of an idealization of explanation as a process of selection, translation, structuring, and realization.

> **Good Explanation.** A good explanation provides information which the interlocutor wants or needs to achieve his or her purpose in asking, as well as information which the explainer has good reason to impose on the interlocutor. It utilizes terminology and concepts the interlocutor is familiar with, or uses the same to construct new terms and concepts which will be needed to express the primary communication. The explanation is structured so that that the interlocutor is conceptually ready to absorb the next proposition, for example, by only requiring that the interlocutor add to or modify his or her beliefs in small increments; motivating each major restructuring of knowledge or new line of reasoning with an example showing why it is needed; and exploiting the ways in which the knowledge is used, or otherwise following common sense orderings in the presentation. Complexity is controlled by keeping justifications or chains of reasoning only a few levels deep. Supporting explanations are separated from the main explanation where necessary to achieve this. Unnecessary detail is avoided. Shift of focus is minimized and organized, finishing topics before leaving them. Questions or requests just posed are answered or acted on before initiating a new agenda. The

> explanation will have greater appeal or impact if it is motivated beforehand, points made are argued for in several ways, and competing points are discredited. Finally, the communicative goals and tactics used to achieve them are adjusted dynamically according to interlocutor feedback.

An Idealized Explanation Process. The remainder of this section attempts to recast these prescriptions within an idealized computational framework. Explanation is characterized as a process of selecting and translating the system's knowledge structures into other structures which specify an explanation, i.e. a plan, and realizing this plan in some communication medium. Being specific about the components of such a process makes it possible to define constraints on explanation by identifying each constraint's origin and how it impacts on this process. Underconstrained aspects of the process suggest areas where further research may identify constraints from previously unknown sources, and provide some freedom for experimentation and use of feedback.

Figure 6.1 illustrates the explanation process, and summarizes discussion to follow. The left column names the sources of constraints on explanation, organized into levels of analysis at which they are identified in an explanation. The middle column lists the constraints themselves. Lines from the left to the middle column indicate the source of each constraint. The right column contains general subprocesses of an explanation process, and lines from the constraints to these indicate which subprocess each constraint impacts on.

The subprocesses of the explanation process are as follows. **Selection** identifies and requests knowledge relevant to the current explanatory goal. The only ordering that occurs at this point is a by-product of shifts in such goals. Selection requires identification of the topic of interest and what kind of knowledge about it is needed, including the desired perspective. These parameters are provided to a retriever which provides requested knowledge, represented in an appropriate conceptual framework (Suthers, 1988a,b). The retriever is in part a **translation** component which enables perspective shifts, and also provides an abstraction barrier between the explainer and the application. The result of selection and retrieval is roughly at the granularity of McKeown's (1982) "relevant knowledge pool" (section 5.1.1) or Souther, Acker, Lester, & Porter's (1989) "viewpoints". **Structuring** organizes the translated material into an explanation plan by **linearizing** the knowledge pool and performing **local coherence** operations on the evolving partial order. This might generate further explanatory subgoals as needed to fill "gaps" in the explanation, resulting in selection, retrieval, and insertion of additional material in the evolving plan. All of these processes annotate their choices about explanation content and structure with the goals and assumptions behind those choices. The resulting plan directs **realization** of the explanation in an appropriate medium, e.g. by generating text and graphics. Annotations on the plan are used to choose medium-specific devices such as illocutionary speech acts. The process is similar to Bienkowski's (1986) "minimal architecture for extemporaneous elaboration", though I sequence the components differently, and leave open the possibility that iterative or simultaneous processing will be necessary to maximally satisfy interacting constraints.

The explanation plan which this process constructs has structure and annotations at levels corresponding to all of the constraints on explanation to be discussed below. These levels are somewhat orthogonal to the selection, translation, structuring, and realization processes just described; e.g. structuring may be embodied in plan operators expressing conceptual, epistemological, and psycholinguistic constraints on the explanation, so structural annotations may occur at each of these levels. The plan is to be distinguished from its physical presentation to the interlocutor. In particular, the

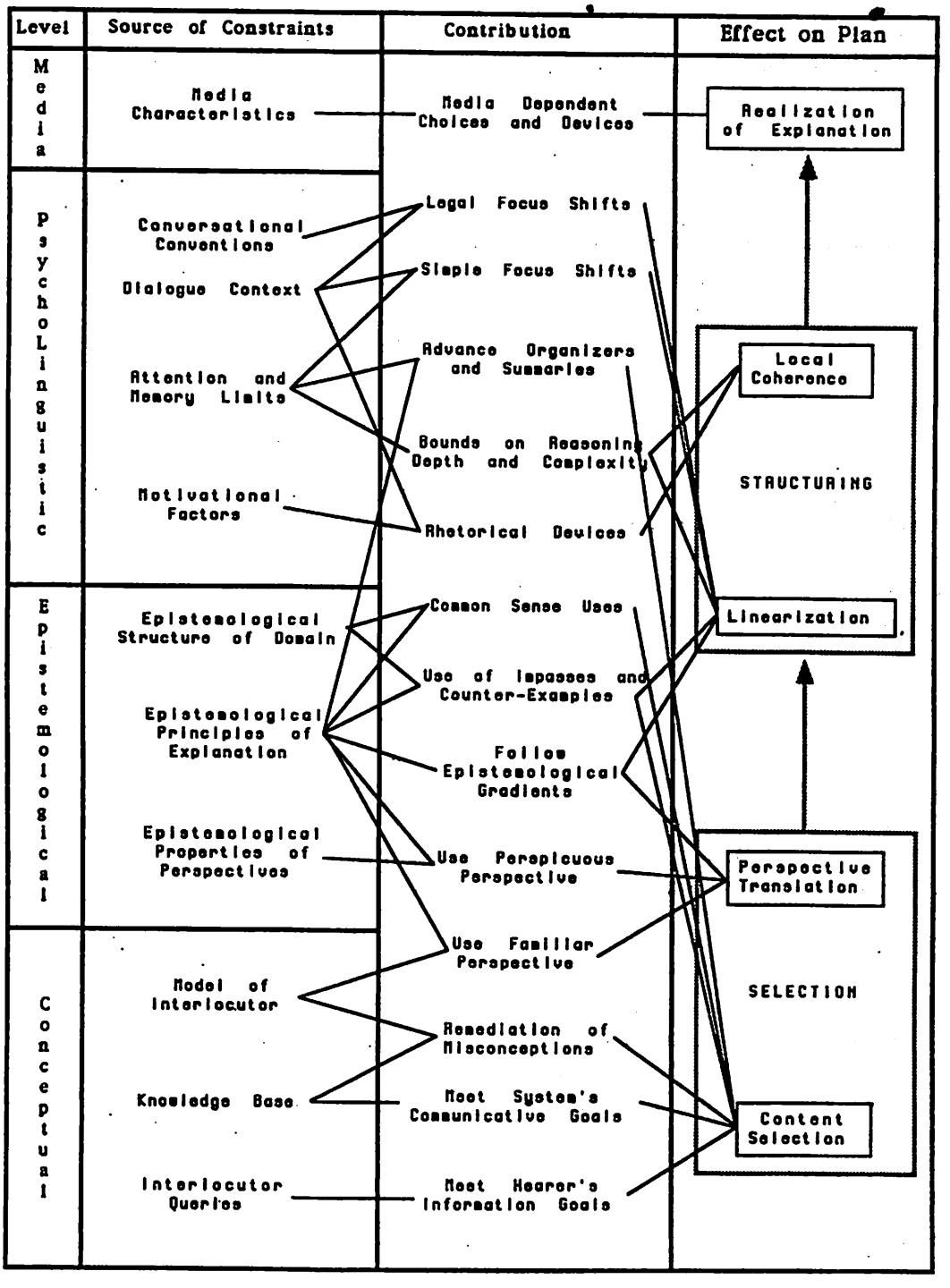


Figure 6.1: Constraints on an Explanation Process

plan may include content which is communicated implicitly, e.g. by relational propositions (Mann & Thompson, 1986) or other inferences the explainer expects the hearer to make (Lehnert, 1984). I am deliberately uncommittal about whether the plan is at a proposition, sentence, or paragraph granularity, and take the (controversial) position that the generation specialist should have the responsibility for deciding what chunks to present in these media-dependent units.

Each subsection below begins with part of the English description of a good explanation. I discuss how the prescription constitutes a constraint on explanation by identifying where the constraint comes from and how it impacts on the idealized process. References to papers and to sections of this literature review are made in footnotes, to avoid cluttering the discussion.

6.1.1 Informative and Conceptual Constraints

> A good explanation provides information which the interlocutor wants or needs to achieve his or her purpose in asking, as well as information which the explainer has good reason to impose on the interlocutor.

Most computer-based systems simply use a fixed mapping of interface events (e.g., a menu selection or question-type) to procedures for retrieving the desired content.[1] More generally, direct constraints which the interlocutor's query places on the informative content of the explanation may be derived from examination of a conceptual representation of the query to identify empty portions of the conceptualization.[2] The result may identify a single piece of information to be conveyed, or may only restrict content selection to a "relevant knowledge pool",[3] some subset of the system's knowledge structures. If necessary, assumptions about the interlocutor (i.e., a user model) may be used to reduce this pool to a manageable size, by filtering what the interlocutor is believed to know[4] or doesn't want to know.[5] Not all the selected information need be communicated explicitly, if the explainer knows what the interlocutor will be able to infer from other information,[6] or can find other ways to communicate it implicitly.[7] Hence, informative constraints do not unequivocally dictate the explicit content of an explanation.

Several elaborations on the derivation of informative constraints are possible. In some contexts, plan recognition and goal inference may be required to identify further information the interlocutor needs.[8] The explainer may have informative goals of its own (e.g. to communicate important results), which have to be balanced against the interlocutor's. Examination of the relationship between the interlocutor's goals and the explainer's knowledge structures can indicate whether and how the explainer can meet these goals. If the interlocutor's request exhibits misconceptions manifested in a mismatch to the explainer's knowledge structures, remediation goals may result.[9]

---

[1] Question types were discussed in section 2.2.3; similar approaches were taken in Weiner's (1980) BLAH (section 2.3.2); Davis, Buchanan, and Shortliffe's (1977) MYCIN (section 2.3.3); and McKeown's (1982, 1985) TEXT (section 5.1.1).

[2] As in Lehnert (1977), section 2.2.3.

[3] McKeown (1982, 1985), section 5.1.1.

[4] Weiner's (1980) BLAH; Cohen & Jones (1987); section 4.1.3.

[5] XPLAIN's viewpoints, section 3.1.2.

[6] Lehnert (1984), section 4.1.3.

[7] E.g. via relational propositions, Mann & Thompson (1986), section 4.2.1.

[8] Allen & Perrault (1980), Carberry (1983), section 4.1.3.

[9] McCoy (1985, 1986), section 5.1.2.

6.1.2 Epistemological Constraints

These come from an examination of knowledge and its use, i.e. analysis of the structure of a domain's knowledge, and theories of how people use and modify their knowledge. I identify two aspects of the explanation process which such considerations bear on: selection of an appropriate perspective, and ordering and augmenting the contents of the relevant knowledge pool within this perspective.

Perspective.

> It utilizes terminology and concepts the interlocutor is familiar with, or uses the same to construct new terms and concepts which will be needed to express the primary communication.

The constraints on choice of perspective[10] have two origins. Examination of the user model suggests what conceptual framework the user is most likely to understand or benefit from.[11] The explainer must also consider which conceptual framework perspicuously expresses the desired content.[12] If perspicuity conflicts with and over-rules familiarity, then subgoals for explaining new terms and concepts may be necessary to familiarize the user with the perspective chosen for the primary communication. Perspective selects content along dimensions intersecting with the conceptual specification of the topic,[13] further refining the relevant knowledge pool. Specialists which know how to translate between these perspectives may be required. The structural primitives of the chosen perspective may constrain selection and linearization of content. This suggests that all ordering activities should take place after translation, in the target perspective. However, perspective may change during an explanation (along with the informative goals) as a function of the content and the discourse context: this is a coarse form of linearization.

Epistemological Structure.

> The explanation is structured so that that the interlocutor is conceptually ready to absorb the next proposition, for example, by only requiring that the interlocutor add to or modify his or her beliefs in small increments;[14] motivating each major restructuring of knowledge or new line of reasoning with an example showing why it is needed;[15] and exploiting the ways in which the knowledge is used, or otherwise following common sense orderings in the presentation.[16]

This portion of the English prescription for a good explanation is also concerned with epistemological criteria, but impacts on the linearization of the relevant knowledge pool, and may generate goals for additional content.[17] The linearization is constrained by deriving a partial ordering from the structure of the knowledge to be expressed, and by exploiting relations between this knowledge and what is presumed about interlocutor beliefs. In the process, the content is augmented where needed

[10] Section 3.2.

[11] Section 4.1.3.

[12] Davis (1984), Stevens & Collins (1980), Stevens & Steinberg (1981), McKeown, Wish, & Matthews (1985), Falkenhainer & Forbus (1888), section 3.2.

[13] Suthers (1988a,b).

[14] Winston's (1975) near misses, Goldstein's (1979) genetic graphs, vanLehn's (1987) felicity conditions, Micro-model evolution as expressed by Stevens & Collins (1980) and White & Frederiksen (1987); section 4.1.

[15] Rissland (1978a,b, 1984), section 4.1.2.

[16] Bienkowski (1986), Rissland (1978a,b), section 4.1.1; Paris & McKeown (1986), section 5.1.3.

[17] References in the previous footnotes, discussion throughout section 4.1.

to help the hearer bridge epistemological "gaps" in the evolving linearization. That the ordering is *partial* is significant: this leaves some freedom for non-epistemological constraints on explanatory structure to operate. Another epistemological issue is the impact of consistency and coherence of explanations on the interlocutor's ability to integrate them into a coherent conceptualization.[18]

6.1.3 Psycho-Linguistic Considerations

While there is no clear line between what is linguistic and what is not, the following considerations are derived in part from linguistic research, and impact on the explanation plan in part as directives for a realization component, e.g. a text planner. The partial ordering resulting from the epistemological constraints is re-arranged and further constrained once the selected content and initial ordering become specific enough for heuristics at this level to operate.

Structural Complexity.

> Complexity is controlled by keeping justifications or chains of reasoning only a few levels deep. Supporting explanations are separated from the main explanation where necessary to achieve this. Unnecessary detail is avoided.

These constraints arise from human attentional and memory limitations. Some of the constraints of the previous section come from this source as well; however, unlike epistemological constraints, structural constraints are *knowledge-independent* limits on structural and logical complexity such as the depth of nested chains of reasoning, the number of things presented at once, etc.[19]

Dialogue Context: Illocutionary Structure and Focus of Attention.

> Shift of focus is minimized and organized, finishing topics before leaving them. Questions or requests just posed are answered or acted on before initiating a new agenda.

Explanation is sensitive to the dialogue context via conversational conventions[20] governing the use of speech acts and shifts in focus of attention. These constrain the generation of explanatory goals and the linearization of content.

The dialogue can be analyzed to identify unfulfilled commitments implied by previous speech acts. This gives constrains which subsequent illocutionary acts will be considered appropriate.[21] Speech acts are defined in part according to whether information is being given or requested; what the explainer's attitude is towards the propositional content; the relationship between the interlocutor's beliefs (whether the explainer is contradicting or agreeing with the other interlocutor); and tense and modality considerations. Working backwards, constraints at an illocutionary level translate into constraints on appropriate explanatory goals. The illocutionary aspect of the goal under which a given conceptual unit was selected is recorded as an annotation on that unit, so that the text generator may realize its content using an appropriate speech act.[22]

[18] White & Frederiksen (1977), Thagard (1988), section 4.1.5.
[19] Chester's (1976) EXPOUND, section 2.3.1, and Weiner's (1980) BLAH, section 2.3.2.
[20] E.g. Grice's (1975) Maxims, section 2.2.1.
[21] Austin (1962), Searle (1969), Bach & Harnish (1979), Winograd (1988), section 2.2.2.
[22] Appelt (1985), section 5.2.1; McCoy (1985, 1986), section 5.1.2.

A simple way to organize focus of attention is to follow a stack discipline.[23] People violate this all the time, but that doesn't necessarily reduce its utility for generating comprehensible explanations. Violations should have good reason, e.g. when a sub-plan is discarded due to discovery of faulty assumptions about the interlocutor, active topics may be thrown away without completion.[24] To meet the previous constraint on structural complexity, it may be necessary to introduce a topic which may be addressed later, in a supporting explanation.[25] This can be done in a stack discipline, but appears to violate the requirement that topics be finished once introduced. However, this depends on a definition of what constitutes a change in focus. More sophisticated definitions consider the illocutionary aspect of "conversational moves", as well as shifts in focal content.[26]

Rhetorical Structure.

> The explanation will have greater appeal or impact if it is motivated beforehand, points made are argued for in several ways, and competing points are discredited.

Here I refer to "purely rhetorical" techniques,[27] and assume that conceptual and epistemological considerations have been factored out.[28] Such techniques exploit human attentional and judgmental characteristics. They can be invoked where degrees of freedom remain in the explanation process. In a purely argumentative or persuasive domain such as legal reasoning, rhetorical techniques would operate earlier in the process, with a stronger influence on the result.[29] Rhetorical devices chosen are recorded at their own level to be expressed by the realization component, and may impact on choice and organization of content as well, e.g. a "convergence" strategy would invoke a subgoal to find an additional explanation supporting a point which has already been explained one way.

Relational propositions[30] may be used to implicitly express some of the planned content through the rhetorical structure of the text. This reduction of explicit content relies on a model of likely interlocutor inferences,[31] and may help meet structural complexity goals.

6.1.4 Feedback and Dynamic Planning

> Finally, the communicative goals and tactics used to achieve them are adjusted dynamically according to interlocutor feedback.[32]

The explanation plan will be underconstrained where there wasn't enough information to choose between alternatives; explanation problem solving ran out of resources; or the planner encountered dubious assumptions. In the latter case, the explainer could allow and record the assumption, taking a different approach at this point if user feedback suggested that the plan failed.[33] Otherwise, a record of the available alternatives could be made, and planning deferred until more information is

[23] McKeown (1982), Grosz (1977), section 2.2.1.
[24] Moore & Swartout, section 5.2.2.
[25] Weiner (1980), section 2.3.2.
[26] Reichman-Adar (1984); also a "view" (Suthers, 1988a,b) might be the right granularity for this.
[27] Dicks (1981) and Rissland (1984b), section 4.2.2.
[28] Section 4.2.1 discusses the need for this factoring.
[29] Ashley (1987), Dicks (1981), Rissland (1984b), section 4.2.2.
[30] Mann & Thompson (1986), section 4.2.1.
[31] Section 4.1.3.
[32] Woolf & McDonald (1984).
[33] Moore & Swartout (1989); section 5.2.3.

available. The plan could include explicit solicitations of feedback to enable making the choice.[34] Planning should be deferred at some threshold of amount and complexity of information into the explanation, since planning too far ahead may assume too much about the effects of the previous explanation on the user's knowledge state.

6.2 Epistemological Research Issues

I believe that the most pressing research issues in explanation have to do with how **epistemological sources of constraints**, viz:

- the internal structure of a domain's shared body of knowledge,
- the role of an individual's knowledge in understanding new concepts and situations, and
- the ways in which individuals are willing or able to transform this knowledge

constrain the selection, organization, and presentation of the content of explanations. The underlying intuitions are that people actively construct their knowledge in an attempt to make sense of their experiences; there are constraints on this construction; and hence on communications intended to aid in the process of understanding. People cannot be expected to comprehend arbitrary information at a given time. Understanding requires compatible experiences and connections with existing concepts. Hence, explanation is more likely to succeed if it follows what I will call "epistemological gradients" along which understanding occurs most easily. I am concerned with how the structure of explanation seeks to follow these gradients. The literature I have reviewed provides some starts, but the various efforts are not well integrated.

The primary research issue might be stated: what **epistemological principles** (constraints and heuristics) guide the structuring of a good explanation? To break this down, the research community could seek principles providing guidance in solving the following interdependent **subproblems of explanation:**

- Identify or be able to generate as needed a "relevant knowledge pool", a subset of the knowledge base which contains material directly relevant to a given communicative goal.
- Using the explainer's partial and approximate model of interlocutor beliefs, make plausible assumptions concerning which components of the relevant knowledge pool are likely to be obvious to the interlocutor, and which are likely to be unintelligible, needing further explaining in themselves.
- Choose a perspective, i.e. a conceptual framework for viewing the topic at hand, which perspicuously expresses the material to be communicated in a way the interlocutor is comfortable with; or introduce a new perspective, so that the interlocutor may understand subsequent communications within that new perspective.
- Order propositions to be communicated to maximize their intelligibility. There are two fundamental aspects of this subproblem: knowing when something is a *prerequisite* for understanding something else, and when and how the *relevance* of something to be communicated is motivated. These issues can only be addressed within the context of a given perspective, since a change in perspective potentially constitutes a change in the primitives in which the content of an explanation is expressed and structured.

[34] Roth, Mattis, & Mesnard (1988) discuss identifying when the explanation is underconstrained and inserting "pauses".

- Determine the need for additional material, including examples and analogies, to fill "epistemological gaps" in the above ordering, where the interlocutor may otherwise fail to incorporate material subsequent to the "gap". For example, such gaps may occur where the interlocutor plausibly lacks prerequisite terminology or concepts, or may not have the experience needed to motivate the relevance of material to follow.

Examples of epistemological principles, one from each of the epistemological sources of constraints (page 88), are: "Follow the natural structure of knowledge"; "Use concepts the interlocutor is familiar with"; and "Expect the interlocutor to modify his/her knowledge in minimal increments." These are extremely general and unconditional guidelines, which further research should progressively make more specific as they are operationalized within the context of some computational approach to generating explanations (e.g., a planning system). When operationalized, these principles will be sensitive to explanatory goals, dialogue context, and the interlocutor's background knowledge, and will be expressed in terms of abstractions describing particular types of knowledge. Hence the operationalized principles will make assumptions about the available domain knowledge and contextual information.

In developing a principled solution to the above subgoals, a variety of related questions will be encountered. I group these into three **constellations of questions**, named according to what type of entity is examined for it role in the structure of explanation:

Conceptual Relations. How can an explainer exploit relations (e.g. generalization, specialization, causation, components, analogy) between concepts to structure an explanation in a manner which provides useful connections to the interlocutor's knowledge and facilitates understanding of new material? How could the undifferentiated "prerequisite" relation so common in tutorial systems be derived from other relations between concepts?

Perspectives. What conceptual frameworks are useful in explanation, and what roles do they play? How does one choose a perspective or move between perspectives to facilitate the interlocutor's understanding? For example, when would one use structural, functional, or causal perspectives on a physical process, and when are analogies appropriate? Or how does the use of implicit assumptions help generate useful simplifications, and how does one elaborate a simplified explanation into a more sophisticated one? What abstractions describe the explanatory properties of perspectives in a useful way?

Examples. What are the roles of examples in the epistemological structure of explanation? By what principles do we decide that an example is needed (e.g. to uncover a hidden assumption, or show where a result fails to generalize), and how do we specify the attributes of the needed example in a general way (rather than using pre-coded indices to stored examples)?

Clearly, much work is required. Fortunately, most of these questions are currently being pursued, though unfortunately the researchers are scattered throughout disciplines and do not share a common theoretical framework within which to integrate their work towards a theory of explanation. I believe it would help if the various epistemological constraints on explanation were made explicit and operationalized within a planning framework where their interactions and integration with the other constraints of section 6.1 could be studied. This is the goal of research I am initiating at the time of this writing.

6.3 Expressing a Theory of Explanation

Here I comment on ways in which a theory of explanation could be expressed, and on some implications for the design of knowledge communication systems.

6.3.1 The Relationship Between Theory and Architecture

In section 6.1, I painted a picture of explanation as a selection, translation, structuring, and realization process which attempts to satisfy a variety of interacting constraints. An initial attempt to integrate all these considerations could utilize a blackboard architecture paralleling figure 6.1. The inherent uncertainty in explanation planning and the need to utilize feedback and react to interruptions suggests that the procedures operating on this blackboard will do so in a dynamic, reactive manner.

However, the open-endedness of such an architecture is unsettling for two reasons. It may be inefficient, and it seems to constitute a lack of theoretical committment towards what kind of process explanation is. David McDonald has argued[35] that without architectural commitments, there is no theory of language generation. Human performance suggests that language generation is highly specialized and mostly a pipelined process. A similar attitude could be taken towards explanation, though at this point we may not understand it well enough to make the kinds of commitments McDonald would ask of us, and may need an open-ended architecture for experimental purposes. In such an approach, the theory is not about architecture, but rather about abstract principles. A uniform architecture which reasons explicitly from these principles is more helpful in developing such principles than one where the principles are embedded in the architecture via decisions made during its design. On the other hand, both architectural plausibility and efficiency seem to require such an embedding of theory in design.

It is often easier in the early phases of a program of artificial intelligence research to get something that "works" first, and then analyze the result to extract the principles behind its success and which explain its failures. With respect to architecture, research goes full cycle from implicitly embedded principles to making them explicit in a more general architecture, and back to re-embedding them into a more efficient architecture which itself constitutes a stronger theoretical committment than the original "first hacks". Explanation research is currently on the first half of this cycle, making implicitly embedded principles explicit. I have already argued that a planning framework has advantages for this phase in the development of a theory of explanation, and will not belabor the point here (see sections 5.1.5 and 5.2).

6.3.2 Abstractions for a Theoretical Vocabulary

Throughout the paper I have criticized previous research for failing to express an explicit theory of explanation which is separable from the implementation details of the system it is embodied in. One remedy has just been discussed: to make the theory explicit in a unified planning framework rather than implicit in a variety of mechanisms. Yet this is just part of the solution: one must still write planning operators which clearly express the theory. The vocabulary of primitive terms used

[35] At the AAAI-88 workshop on Text Generation.

to write these operators affect the clarity and generality of the theory. Here I list various kinds of abstractions which may be useful in developing an appropriate vocabulary. These roughly parallel the levels of section 6.1.

Semantic/Conceptual abstractions describe conceptual primitives (e.g., predicates and relations) in terms of the kinds of knowledge they make available. Abstracting explanatory content from the particular predicates of a domain promotes the generality of a theory, and facilitates portability of an implementation between domains and between different representational formalisms.[36]

Epistemological abstractions, such as epistemological classes, dual relations, and other relations between knowledge along which new knowledge is connected to the familiar, can help express theories of how to choose and order content.[37] Epistemological and conceptual abstractions may overlap, especially where the epistemological theory describes how to choose an appropriate perspective,[38] and how to exploit structure at the conceptual level.[39] This is new ground for artificial intelligence research; other disciplines concerned with pedagogical issues should be consulted.

Logical and Syntactic abstractions describe content-independent, properties of representations themselves, such as the length of an expression, depth and branching of a tree, or depth of a stack. These properties have been manipulated for a variety of purposes, e.g. to reduce complexity and increase the comprehensibility of an explanation,[40] control focus of attention when linearizing content,[41] and to choose target skills which make incremental modifications to the interlocutor's skills.[42] Syntactic abstractions apply to the data structures of diverse architectures, and are somewhat independent of the content being communicated, yet carry with them some dangers. Heuristics which are expressed in syntactic terms don't state the theoretical considerations that went into them. Logical and syntactic abstractions provide leverage for formal analysis, but also make it possible to consider abstract cases which may not occur in reality.

Illocutionary abstractions help express explanatory goals,[43] and enable annotations which constrain choices made in the realization of the explanation, i.e. to convey rhetorical force.[44] Speech acts, via the commitments they imply, also help evaluate the appropriateness of communicative acts within a given discourse context.[45]

Rhetorical Predicates and Argument Tactics require further analysis. As currently conceived, abstractions from the other categories are included along with purely rhetorical and psychological considerations.[46]

I believe that attention to design of appropriate abstractions will facilitate both machine reasoning about explanation, and communication between researchers concerning computational theories

[36] See Gilbert (1987) and Suthers (1988a,b), and the various ways in which the conceptual contributions of perspectives are characterized in section 3.2.

[37] Rissland (1978a,b), section 4.1.

[38] Section 3.2.

[39] Section 4.1.1.

[40] Weiner (1980), section 2.3.2.

[41] Paris & McKeown (1986), section 5.1.3.

[42] Goldstein (1979), section 4.1.1.

[43] Appelt (1985) does this.

[44] Section 2.2.2, also McCoy (1985, 1986), section 5.1.2.

[45] Winograd (1988), section 2.2.2.

[46] Sections 4.2.1 and 4.2.2.

of such reasoning. Appropriate abstractions separate theory from implementation, enabling us to express our theories without describing the details of how a program works, and enable application of a theory beyond the domains in which it was developed. A theory of explanation which is less dependent on particular architectures and data structures may be implemented in a more portable manner.

6.3.3 Coupling: Implications for KBS Design

Recall that the coupling problem has to do with the ways in which design of explanation and problem solving facilities impact on each other. (I have discussed the coupling between a theory of explanation and architecture in section 6.3.1). My conclusions concerning the desirability of coupling are as follows. Coupling is almost always a good thing at design time, in that a knowledge-based system which is to serve as a communication medium must be *designed around the path along which knowledge travels.* Design of knowledge acquisition, application problem solving, and explanation "modules" must be coordinated. With respect to the knowledge base, several comments may be made. The representation must be designed to be epistemologically adequate for explanation, and may require representation of information not required for problem solving. Consistency and nonredundancy dictate a shared knowledge base, but modularity and freedom to design client programs as needed dictates customized representations. The solution is as in database systems: build a centralized knowledge base, but allow virtual "views" of it (Suthers 1988a,b). During knowledge acquisition, the knowledge engineer must have the communicative goals of the system in mind, and be required to supply the knowledge which supports these goals. The problem solver must be able to connect the problem solving trace to justifying knowledge. At run time, decoupling is a good thing when the application problem solver needs to run unencumbered, and the explainer needs to be able to bring the user towards understanding the results in a manner suitable for humans. Exceptions to run time decoupling are:

- When knowledge communication is an equal or the primary purpose of the system, it may be appropriate for explanatory goals to direct problem solving activities towards those which produce results relevant to those goals.
- The validity of the explanation must stand and fall with that of the problem solving in a manner allowing it to serve as a basis for judging the recommended conclusions. This holds even more in domains where there are external standards of correctness; and less in rhetorical domains such as legal reasoning.

In general, issues of knowledge communication are not localized, say, to an "explanation module" tacked onto an existing system. To achieve explanatory goals, one must design epistemologically principled architectures to support the interactive communication process.

References

Allen, J. F. & Perrault, C. R. (1980). Analyzing intention in utterances. *Artificial Intelligence,* vol. 15, pp. 143-178.

Appelt, D. E. (1985). Planning English referring expressions. *Artificial Intelligence,* vol. 26, no. 1, pp. 1-33.

Ashley, K. (1987). Modelling Legal Argument: Reasoning with Cases and Hypotheticals. *PhD Dissertation, University of Massachusetts.*

Auramaki, E., Lehtinen, E., & Lyytinen, K. (1988). A speech-act-based office modeling approach. *ACM Transactions on Office Information Systems,* vol. 6, no. 2, pp. 126-152.

Austin, J. L. (1962). *How to Do Things with Words.* Cambridge, Massachusetts: Harvard University Press.

Bach, K., & Harnish, R. M. (1979). *Linguistic Communication and Speech Acts.* Cambridge, Massachusetts: MIT Press.

Bienkowski, M. A. (1986). A Computational Model for Extemporaneous Elaborations. *CSL Report 1,* Cognitive Science Laboratory, Princeton University, Princeton, N.J.

Birnbaum, L. (1982). Argument molecules: A functional representation of argument structure. *Proc. 3rd AAAI,* Pittsburgh, PA, pp. 63-65.

Birnbaum, L., Flowers, M., & McGuire, R. (1980). Towards an AI model of argumentation. *Proc. 1st AAAI,* Stanford, CA, pp. 313-315.

Bobrow, D. G. & Winograd, T. (1977). An overview of KRL, a knowledge representation language. *Cognitive Science,* vol. 1, pp. 3-46.

Brown, J. S.; Burton, R. R.; & Zydel, F. (1973). A model-driven question-answering system for mixed-initiative computer-assisted instruction. *IEEE Transactions on Systems, Man, and Cybernetics,* vol. 3, pp. 248-257.

Brown, J. S.; & Burton, R. R. (1975). Multiple representation of knowledge for tutorial reasoning. In Bobrow, D. & Collins, A. (Eds.) *Representation and Understanding: Studies in Cognitive Science.* Academic Press, New York.

Brown, J. S.; & Burton, R. R. (1978). Diagnostic models for procedural bugs in basic mathematical skills. *Cognitive Science,* vol. 2, pp. 155-192.

Burton, R. R. & Brown, J. S. (1979a). Toward a natural language capability for computer-assisted instruction. In O'Neil, H. (Ed.) *Procedures for Instructional Systems Development.* Academic Press, New York.

Burton, R. R. & Brown, J. S. (1979b). An investigation of computer coaching for informal learning activities. *International Journal of Man-Machine Studies,* vol. 11, pp. 5-24. (Reprinted in Sleeman, D. H., & Brown, J. S. (Eds.) *Intelligent Tutoring Systems.* Academic Press, London.)

Carr, B.; & Goldstein, I. P. (1977). Overlays: a theory of modeling for computer-aided instruction. *AI Lab Memo 406 (Logo Memo 40).* Massachusetts Institute of Technology, Cambridge, Mass.

Carberry, S. (1983). Tracking user goals in an information-seeking environment. *Proc. AAAI-83,* pp. 59-63.

Carbonell, J. R. (1970). AI in CAI: an artificial intelligence approach to computer-assisted instrution. *IEEE Transactions on Man-Machine Systems,* vol. 11, no. 4, pp. 190-202.

Cawsey, A. (1988). Explaining the behavior of simple electronic circuits. *Proc. Intelligent Tutoring Systems,* June 1-3, Montreal, pp. 372-378.

Chandrasekaran, B., Tanner, M., & Josephson, J. (1989). Explaining control strategies in problem solving. *IEEE Expert,* Spring 1989, pp. 9-24.

Chester, D. (1976). The translation of formal proofs into English. *Artificial Intelligence,* vol. 7, pp. 261-278.

Clancey, W. J. (1979). Dialogue management for rule-based tutorial. *Proc. 6th IJCAI,* Tokyo, pp. 155-161.

Clancey, W. J. (1983). The epistemology of a rule-based expert system – a framework for explanation. *Artificial Intelligence,* vol. 20, no. 3, pp. 215-251.

Clancey, W. J. (1986a). From GUIDON to NEOMYCIN and HERACLES in twenty short lessons. *AI Magazine,* vol. 7, no. 3, pp. 40-60.

Clancey, W. J. (1986b). Qualitative student models. *Annual Review of Computer Science,* vol 1., pp. 381-450.

Clancey, W.J., & Lestinger, R. (1981). NEOMYCIN: Reconfiguring a rule-based expert system for application to teaching. *IJCAI 7* , Vancouver, pp. 829-836.

Clement, J. J. (1978). Some types of knowledge used in understanding physics. *Cognitive Processes Research Group report,* University of Massachusetts, Amherst, MA.

Clement, J. J. (1988). Observed methods for generating analogies in scientific problem solving. *Cognitive Science,* vol. 12, no. 4, pp. 563-586.

Cohen, R. (1988). Producing user-specific explanations. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 44-47.

Cohen, R. & Jones, M. (1987). Incorporating user models into an expert system for educational diagnosis. In *User Models in Dialog Systems.* W. Wahlster & A. Kobsa eds., Springer-Verlag. Also available as *Research Report CS-86-37.* University of Waterloo, Canada,

Cohen, R. & Perrault (1979). Elements of a plan-based theory of speech acts. *Cognitive Science,* vol. 3, pp. 177-212.

Collins, A. (1977). Processes in acquiring knowledge. In Anderson, R. C.; Spiro, R. J.; & Montague, W. E. (Eds.), *Schooling and the Acquisition of Knowledge.* LEA: Hillsdale.

Collins, A. (1978). Fragments of a theory of human plausible reasoning. In D. L. Waltz (Ed.) *Theoretical Issues in Natural Language Processing 2.* Urbana-Champaign: University of Illinois.

Collins, A. (1985). Component models of physical systems; *Proc. 7th Annual Conference of the Cognitive Science Society,* Irvine, Cal. pp. 80-89.

Collins, A. & Gentner (1983). Multiple models of evaporation processes. *Proc. 5th Annual Conference of the Cognitive Science Society,*

Collins, A., Warnock, E. A., Aiello, N., & Miller, M. L. (1975). Reasoning from incomplete knowledge. In D. G. Bobrow & A. Collins (Eds.) *Representation and Understanding.* New York: Academic Press.

Davis, R. (1979). Interactive transfer of expertise: Acqusition of new inference rules. *Artificial Intelligence,* vol. 12, pp. 121-157.

Davis, R. (1984). Diagnostic Reasoning based on Structure and Behavior. *Artificial Intelligence,* vol. 24, pp. 347-410.

Davis, R., Buchanan, B. G., & Shortliffe, E. H. (1977). Production rules as a representation for a knowledge-based consultation program. *Artificial Intelligence,* vol. 8, no. 1, pp. 15-45.

deJong, G. & Mooney, R. (1986). Explanation-based learning: An alternative view. *Machine Learning,* vol. 1, no. 2, pp. 145-176.

de Kleer, J. & Brown, J. S. (1980). Mental models of physical mechanisms. *CIS-3,* Cognitive and Instructional Sciences Series, Xerox Palo Alto Research Center.

de Kleer, J. & Brown, J. S. (1983). Assumptions and ambiguities in mechanistic mental models. In *Mental Models,* D. Gentner & A. Stevens (Eds.); Hillsdale: LEA.

Dhar, V. & Pople, H. (1987). Rule-based versus structure-based models for explaining and generating expert behavior. *Communications of the ACM,* vol. 30, no. 6, pp. 542-555.

Dicks, V. (1981). Courtroom rhetorical strategies: Forensic and deliberative perspectives. *QJS 67* , pp. 178-192.

Eriksson, A. & Johansson, A. (1985). Neat explanation of proof trees. *9th IJCAI,* August 1985, Los Angeles, pp. 379-381.

Falkenhainer, B. & Forbus, K. D. (1988). Setting up large-scale qualitative models. *Proc. AAAI-88,* August 21-26, St. Paul, pp. 301-306.

Frederiksen, J. R. & White, B. Y. (1988). Intelligent learning environments for science education. *Proc. Intelligent Tutoring Systems,* June 1-3, Montreal (ITS-88), pp. 250-257.

Furth, H. G. (1969). *Piaget and Knowledge.* Englewood Cliffs, New Jersy: Prentice Hall Inc.

Gentner, D. & Gentner, D. (1983). Flowing waters or teeming crowds: Mental models of electricity. In *Mental Models,* D. Gentner & A. Stevens (Eds.); Hillsdale: LEA.

Gilbert, G. N. (1987). Question and answer types. *Research and Development in Expert Systems IV,* D.S. Moralee, ed., Cambridge University Press, pp. 162-172.

Gilbert, G. N. (1988). Forms of explanation. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 72-75.

Goldstein, I. (1979). The genetic graph: a representation for the evolution of procedural knowledge. *International Journal of Man-Machine Studies,* vol. 11, pp. 51-77.

Grice, H. P. (1975). Logic and conversation. In Cole & Morgan (Eds.) *Syntax and Semantics,* vol. 3.

Grosz, B. J. (1977). The representation and use of focus in a system for understanding dialogs. *Proc. 5th IJCAI, Cambridge,* pp. 67-76.

Grosz, B. J. & Sidner, C. L. (1985). Discourse structure and the proper treatment of interruptions. *Proc. 9th IJCAI,* Los Angeles, California, August 1985, pp. 832-839.

Harrè, R. (1983). *The logic of the sciences.* New York: St. Martin's Press.

Hasling, D. W.; Clancey, W. J., & Rennels, G. (1984). Strategic explanations for a diagnostic consultation system. *International Journal of Man-Machine Studies,* vol. 20, pp. 3-19

Hewson, P. W. (1981). A conceptual change approach to learning science. *European Journal of Science Education,* vol. 3, no. 4, pp. 383-396.

Hill, W. C. (1989). The mind at AI: Horseless carriage to clock. *AI Magazine,* vol. 10, no. 2, pp. 29-41.

Hobbs, J. R. (1985). Granularity. *Proc. Ninth International Joint Conference on Artificial Intelligence,* Los Angeles, August 1985, pp. 432-435.

Hovy, Eduard H. (1988a). Planning Coherent Multisentential Text; *Proc. 26th Meeting of the ACL, Buffalo, New York;* Reprinted in ISI/RS-88-208, Information Sciences Institute, Marina del Rey, California.

Hovy, Eduard H. (1988b). Approaches to the planning of coherent text. Presented at the *4th International Workshop on Text Generation,* Catalina Island, California, July 1988.

Joshi, A., Webber, B., & Weischedel, R. (1984). Living up to expectations: Computing expert responses. *Proceedins of AAAI-84,* August 6-10, Austin, Texas, pp. 169-175.

Kaplan, J. (1983). Cooperative responses from a portable natural language database query system. In *Computational Models of Discourse.* M. Brady & R. Berwick, eds. Cambridge: MIT Press.

Kass, R. & Finin, T. (1987). Rules for the implicit acquisition of knowledge about the user. *Proc. AAAI-87,* Seattle, pp. 295-300.

Kass, R. & Finin, T. (1988). General user modeling: A facility to support intelligent interaction. *Proc. Architectures for Intelligent Interfaces Workshop,* Monterey, California, March 1988 (ACM/SIGCHI).

Lehnert, W. G. (1977a). Human and computational question answering. *Cognitive Science,* vol. 1, pp. 47-73.

Lehnert, W. G. (1977b). A conceptual theory of question answering. *Proc. Fifth IJCAI,* Cambridge, MA, pp. 158-164. Los Altos: William Kaufmann, Inc.

Lehnert, W. (1984). Problems in Question Answering. in L. Vaina and J Hintikka (eds.), *Cognitive Constraints on Communication,* pp. 137-159.

Lesgold, Alan (1988). Toward a theory of curriculum for use in designing intelligent instructional systems. In Mandl & Lesgold, *Learning Issues for Intelligent Tutoring Systems* , Springer-Verlag, NY, pp. 114-137.

Levi, E. H. (1948). *An Introduction to Legal Reasoning.* The University of Chicago Press.

Linde, C. (1974). *The Linguistic Encoding of Spatial Information.* Doctoral Dissertation, Columbia University.

Mann, W. (1984). Discourse structures for text generation. *Proceedings of the 1984 Coling/ACL Conference,* Stanford CA.

Mann, W. C., & Thompson, S. A. (1983). Relational Propositions in Discourse. *ISI/RR-83-115,* Information Sciences Institute, University of Southern California, Marina del Rey, California.

Mann, W. C., & Thompson, S. A. (1986). Rhetorical structure theory: Description and construction of text structures; *Proceedings of the NATO Advanced Research Workshop on Natural Language Generation,* Nijmegen, The Netherlands, August 19-23, 1986. Also published by Martinus Nijhoff, Dordrecht, 1987.

Maybury, M. (1988). Explanation rhetoric: The rhetorical progression of justifications. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 16-20.

McCoy, K. F. (1985). The role of perspective in responding to property misconceptions; *Proc. 9th IJCAI,* pp. 791-793.

McCoy, K. F. (1986). Contextual effects on responses to misconceptions; *Proceedings of the NATO Advanced Research Workshop on Natural Language Generation,* Nijmegen, The Netherlands, August 19-23, 1986. Also published by Martinus Nijhoff, Dordrecht, 1987.

McKeown, K. R. (1982). *Generating Natural Language Text in Response to Questions about Database Structure.* PhD. Thesis, University of Pennsylvania.

McKeown, K. R. (1985). Discourse strategies for generating natural language text. *Artificial Intelligence,* vol. 27, no. 1, pp. 1-41.

McKeown, K. R., Wish, M. & Matthews, K. (1985). Tailoring explanations for the user. *Proc. 9th IJCAI,* pp. 794-798.

Mitchell, T. M., Keller, R. M., & Kedar-Cabelli, S. T. (1986). Explanation-based generalization: A unifying view. *Machine Learning,* no. 1, vol. 1, pp. 47-80.

Moore, J. D. & Swartout, W. R. (1988). A reactive approach to explanation. Presented at the *Fourth International Workshop on Natural Language Generation, Catalina Island, CA, July 1988.*

Moore, J. D. & Swartout, W. R. (1989). A reactive approach to explanation. *IJCAI-89,* Detroit.

Neches, R., Swartout, W. R., & Moore, J. D. (1985a). Enhanced maintenance and explanation of expert systems through explicit models of their development; *IEEE Transactions on Software Engineering,* vol. SE-11, no. 11, pp. 1337-1351.

Neches. R., Swartout, W. R., & Moore, J. D. (1985b). Explainable (and maintainable) expert systems. *Proc. 9th IJCAI,* August 1985, Los Angeles CA, pp. 382-389.

Paris, C. L. (1985). Description strategies for naive and expert users. *Proceedings of the 23rd Annual Meeting of the Association for Computational Linguistics.* Chicago, 1985.

Paris, C. L. (1987). Combining discourse strategies to generate descriptions to users along a Naive/Expert spectrum. *Proc. 10th IJCAI,* August 1987, Milan, Italy, pp. 626-632.

Paris, C. L. & McKeown, K. R. (1986). Discourse strategies for describing complex physical objects; *Proceedings of the NATO Advanced Research Workshop on Natural Language Generation,* Nijmegen, The Netherlands, August 19-23, 1986. Also published by Martinus Nijhoff, Dordrecht, 1987.

Paris, C. L., Wick, M. R. & Thompson, W. B. (1988). The line of reasoning versus the line of explanation. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 4-7.

Patel, V. L. & Groen, G. J. (1986). Knowledge based solution strategies in medical reasoning. *Cognitive Science,* vol. 10, no. 1, pp. 91-116.

Patil, R. S. (1981). Causal Representation of Patient Illness for Electrolyte and Acid-Base Diagnosis. *TR-267,* Laboratory for Computer Science, MIT.

Piaget, J. (1971). *Genetic Epistemology.* (Translated by E. Duckworth.) New York: W. W. Norton.

Quine, W. V. & Ullian, J. S. (1970). *The Web of Belief.* New York: Random House.

Reichman-Adar, R. (1984). Extended person-machine interface. *Artificial Intelligence,* vol. 22, no. 2, pp. 157-218.

Rich, E. (1981). Users are individuals: individualising user models. *International Journal of Man-Machine Studies,* vol. 18, pp. 199-214.

Rissland, E. L. (1978a). (Formerly Michener.) The structure of mathematical knowledge. *AI Technical Report No. 472,* Massachusetts Institute of Technology.

Rissland, E. L. (1978b). (Formerly Michener.) Understanding Understanding Mathematics. *Cognitive Science,* vol. 2, no. 4.

Rissland, E. L. (1980). Example Generation. *Proc 3rd National Conference of Canadian Society for Computational Studies of Intelligence,* Victoria, BC, pp. 280-288.

Rissland, E. L. (1984a). The ubiquitious dialectic. *Proc 6th European Conference on Artificial Intelligence (ECAI-84),* Pisa, Italy.

Rissland, E. L. (1984b). Argument moves and hypotheticals. *Proceedings First Annual Conference on Law and Technology,* Houston, Texas.

Rissland, E. L., Valcarce, E. M., & Ashley, K. D. (1984). Explaining and Arguing with Examples. *Proc. AAAI-84,* Austin Texas, August 6-10, 1984, pp. 288-294.

Rissland, E. L. & Ashley, K. D. (1986). Hypotheticals as heuristic device. it Proc. AAAI-86, Philadelphia, pp. 289-297.

Rochowiak, D. (1988). Simple explanations and reasoning: From philosophy of science to expert systems. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 95-98.

Roth, S. F., Mattis, J. & Mesnard, X. (1988). Graphics and Natural Language as Components of Automatic Explanation. *Proc. Architectures for Intelligent Interfaces Workshop,* Monterey, California, March 1988 (ACM/SIGCHI), pp. 109-128.

Schank, R. C. (1975). *Conceptual Information Processing.* New York: American Elsevier.

Searle, J. (1969). *Speech Acts.* Cambridge, England: Cambridge University Press.

Self, J. A. (1988). Bypassing the intractable problem of student modeling. *Proc. International Conference on Intelligent Tutoring Systems, Montreal, June 1988,* pp. 18-24.

Sidner, C. (1979). Focusing in the comprehension of definite anaphora. *Computational Models of Discourse,* M. Brady & R. Berwick, eds., pp. 267-330, Cambridge, Mass: MIT Press.

Souther, A. Acker, L., Lester, J., & Porter, B. (1989). Using view types to generate explanations in intelligent tutoring systems. *Proc. Cognitive Science Conf.,* Montreal, 1989.

Stevens, A. & Steinberg, C. (1981). A typology of explanations and its application to intelligent computer aided instruction. *Report No. 4626,* Bolt Beranek and Newman Inc., Cambridge MA.

Stevens, A. L. & Collins, A. (1977). The goal structure of a Socratic tutor. *Proc. National ACM Conference,* Seattle, Washington. pp. 256-263. (Longer version is BBN Report No. 3518.)

Stevens, A. L. & Collins, A. (1980). Multiple conceptual models of a complex system. In R. E. Snow, P. Federico, and W. E. Montague (Eds.), *Aptitude, Learning, and Instruction (Vol. 2).* Hillsdale, NJ: Erlbaum, 1980. pp. 177-197.

Stevens, A. L. Collins, A., & Goldin, S. (1979). Misconceptions in student's understanding. *International Journal of Man-Machine Studies, vol. 11.* pp. 145-156. (Reprinted in Sleeman, D. H., & Brown, J. S. (Eds.) *Intelligent Tutoring Systems.* Academic Press, London.)

Stucky, B. K. (1986). Understanding Legal Argument. *Counselor Project Technical Memo #13.* Department of Computer and Information Science, University of Massachusetts.

Suthers, D. D. (1988a). Providing multiple views of reasoning for explanation. *Proc. International Conference on Intelligent Tutoring Systems,* Montreal, June 1988, pp. 435-442.

Suthers, D. D. (1988b). Providing multiple views for explanation. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 12-15.

Suthers, D. D. & Rissland, E. L. (1988). Constraint manipulation for example generation. *COINS Technical Report 88-71,* Computer and Information Science, University of Massachusetts, Amherst.

Swartout, W. R. (1977). A Digitalis Therapy Advisor with explanations. *Proceedings of the 5th IJCAI,* Cambridge, MA.

Swartout, W. R. (1983). XPLAIN: A system for creating and explaining expert consulting programs. *Artificial Intelligence,* vol. 21, no. 3, pp. 285-325.

Sweller, J. (1988). Cognitive load during problem solving: Effects on learning. *Cognitive Science,* vol. 12, no. 2, pp. 257-285.

Tanner, M. C. & Josephson, J. R. (1988). Justifying diagnostic conclusions. *Proc. AAAI-88 Workshop on Explanation,* St. Paul, August 22, 1988, pp. 76-79.

Taylor, D. M. (1970). *Explanation and Meaning.* Cambridge University Press.

Tennyson, R. & Park, O. (1980). The teaching of concepts: a review of instructional design literature. *Review of Educational Research.* vol. 50, no. 1, pp. 55-70.

Thagard, P. (1988). Explanatory coherence. *Behavioral and Brain Sciences.*

Toulmin, S. (1958). *The Uses of Argument.* Cambridge, England: Cambridge University Press.

Toulmin, S., Rieke, R., & Janik, A. (1979). *An Introduction to Reasoning.* New York: MacMillan.

van Beek, P. G. (1986). A Model for User-Specific Explanations from Expert Systems. *Research Report CS-86-42,* Department of Computer Science, University of Waterloo.

van Beek, P. G. (1987). A model for generating better explanations. *Proc. Association for Computational Linguistics,* pp. 215-220.

vanLehn, K. (1987). Learning one subprocedure per lesson. *Artificial Intelligence,* vol. 31, no. 1, pp. 1-40.

von Glaserfeld, E. (1989). Cognition, construction of knowledge, and teaching. *SYNTHESE* , special issue on Philosophy of Science and Education. Also available as Scientific Reasoning Research Institute report #184, University of Massachusetts, Amherst MA.

Wahlster, W. & Kobsa, A. (1986). Dialogue-based user models. *Proceedings of the IEEE,* vol. 74, no. 7, pp. 948-960.

Weiner, J. L. (1980). BLAH, A system which explains its reasoning. *Artificial Intelligence,* vol. 15, no. 1, pp. 19-48.

Wenger, E. (1987). *Artificial Intelligence and Tutoring Systems: Computational and Cognitive Approaches to the Communication of Knowledge.* Los Altos: Morgan Kaufmann.

White, B. Y. & Frederiksen, J. R. (1987). Causal Model Progressions as a Foundation for Intelligent Learning Environments. *Report No. 6686,* Bolt, Beranek and Newman Inc. Cambridge MA.

Wick, M. R. & Thompson, W. B. (1988). Plausible explanation: Explanation as complex problem solving. *Unpublished draft.*

Wick, M. R. & Thompson, W. B. (1989). Reconstructive explanation: Explanation as complex problem solving, submitted to *IJCAI-89.*

Wilensky, R. (1984). Knowledge Representation - A Critique and A Proposal; *Proceedings of the Sixth Annual Conference of the Cognitive Science Society,* Boulder, Colorado, June 28-30, 1984, pp. 344-357.

Williams, M. D., Hollan, J. D., & Stevens, A. L. (1983). Human reasoning about a simple physical system. In *Mental Models,* D. Gentner & A. Stevens (Eds.). Hillsdale: LEA.

Winkels, R., Breuker, J., & Sandberg, J. (1988). Didactic discourse in intelligent help systems. *Proc. Intelligent Tutoring Systems,* Montreal, June 1988, pp. 279-285.

Winograd, T. (1988). A language /action perspective on the design of cooperative work. *Human Computer Interaction,* vol. 3, no. 1, pp. 3-30.

Winston, P. H. (1975). Learning structural descriptions form examples. In P. H. Winston (Ed.) *The psychology of computer vision.* NY: McGraw Hill.

Woods, W. A., Kaplan, R., & Nash-Webber, B. (1972). The lunar sciences natural language information system: Final report. *BBN Report 2378.* Bolt Beranek and Newman, Inc., Cambridge, Mass.

Woolf, B. & McDonald, D. D. (1984). Building a computer tutor: Design issues; *IEEE Computer,* September 1984, pp. 61-73.