AI Magazine Volume 11 Number 5 (1990) (© AAAI)

Full-Sized Knowledge-Based Systems Research Workshop

Barry G. Silverman and Arthur J. Murray

The Full-Sized Knowledge-Based Systems Research Workshop was held May 7-8, 1990 in Washington, D.C., as part of the AI Systems in Government Conference sponsored by IEEE Computer Society, Mitre Corporation and George Washington University in cooperation with AAAI. The goal of the workshop was to convene an international group of researchers and practitioners to share insights into the problems of building and deploying Full-Sized Knowledge Based Systems (FSKBSs).

The term "full-sized" was chosen to encourage discussion of questions not only of largeness but also of breadth, depth, maturity, and deployment scale. For example, a 1000-rule expert system facilitating knowledge sharing and collaboration between several thousand users was felt to be as interesting to the workshop as a 100,000-rule system with only a few users. That notwithstanding, the underlying question was how to overcome the brittleness and narrowness of the first generation of expert systems, and how to use a variety of new ideas and technologies to increase the scale, intelligence, and capability of the systems currently able to be fielded. Both lessons learned from applications to date and research ideas for future FSKBSs were solicited.

Twenty-five presentations were originally clustered into eight sessions according to perceived themes and to original research questions posed to the participants. Those sessions included: applications, real time distributed systems, representation, knowledge acquisition, tools, natural language, major initiatives and a post-workshop summary session. As the workshop unfolded, discussions focused on five principal topics: (1) practical lessons learned from actual applications, (2) theories for ontological engineering and "knowledge soup", (3) open knowledge server research issues, (4) organization-computer collaboration factors, and (5) national initiatives.

The latter grouping better describes the work presented and is used here as an organizing framework. The descriptions of these five topics have been fed back to the authors and discussants for comment, and to stimulate revision of their papers to more clearly elaborate their concepts, positions and/or lessons learned. We have also invited two papers (reprints) for inclusion in the proceedings that, while not presented at the workshop, were considered integral to the discussion.

Practical Lessons Learned from Deployed FSKBSs

Many of the systems presented at the workshop are currently in active use by a large number of users. This section examines the practical lessons learned from deployed FSKBSs. These include:

1) The Force Requirements Expert System (FRESH) and the Capabilities Assessment Expert System (CASES) of the DARPA/Navy Fleet Command Center Battle Management Program (FCCBMP).
2) Prospector II of the US Geological Survey.
3) The Logistics Planning and Requirements Simplification System (LOGPARS) of the US Army Material Command's Material Readiness Support Activity.
4) The Program Manager's Support System (PMSS) of the Defense Systems Management College.
5) The Thermal Performance Advisor (TPA) of the General Physics Corporation.
6) The Tactical Expert Mission Planner (TEMPLAR) of the US Air Force's Rome Air Development Center.

Digital Equipment Corporation's XCON system was not presented at the workshop, but is included as a basis for lesson's learned due to its status as one of the most well known and long-lived FSKBS, as well as the detailed reports on its development in the technical literature.

Each of the systems mentioned attempts to solve problems in large, multifaceted domains. They automate and assist decision making processes that were previously performed by teams of experts. In that sense, they serve as examples of organization- computer collaboration systems as discussed later. These systems also contain knowledge from multiple sources; facts and procedures obtained from handbooks and manuals are as important as heuristics obtained from experts. LOGPARS was originally intended for use by manufacturing plant personnel to validate the correctness of system orders; it is now used by manufacturing technicians and field service installers for system assembly advice.

The systems are large according to several dimensions. TEMPLAR requires 200MB of hard disk and has 300,000 lines of Lisp code. XCON has 50,000 lines of code in traditional programming languages in addition to over 10,000 rules.

Modular architectures and a system development methodology that blends rapid prototyping with more traditional system engineering techniques characterize many of these systems. FRESH/CASES and XCON both expended significant resources on system engineering issues, and identified system engineers as key members of the development team. Use of diverse languages, knowledge representations, heterogeneous hardware, and mixed-initiative and multimodal man-machine interface styles characterized the FSKBSs. Other system issues such as integration with existing databases, maintenance and version control, testing and evaluation, and documentation also had to be addressed in these systems.

Many system developers discussed the inadequacy of available shells or tools. Ed Feigenbaum challenged them, asserting that "techies" would rather do technical things such as building shells than face the real knowledge engineering work required to build applications, and that the field would progress more rapidly if developers just bought and used off-the-shelf shells and tools. Developers responded with several

points. It was noted that the systems presented at the workshop, selected partially for their degree of maturity, were begun several years ago when available tools and shells were less capable. (Deficiencies from that earlier generation of shells included non-reentrant code, lack of ability to integrate with database systems, and lack of context mechanisms for handling "what-if" reasoning.) It was recognized that shells were useful as rapid prototyping tools, but that as systems grew larger and as the problem domains were better understood, shells tended to be more of a hindrance than a help. Lack of fit between a shell's paradigm and the problem characteristics, and the high overhead of shells especially that associated with unneeded portions, were cited as two difficulties.

From a performance perspective, none of the systems can yet claim to be saving their organizations $40 million/year like Digital's XCON does, but some of the savings are impressive. LOGPARS is in use at 400 sites. It has already been applied to 30 weapon systems saving each of them weeks to months of time. Training time has been reduced 4-fold while output quality and consistency have steadily risen. In a similar fashion TEMPLAR has been used in military exercises since 1988, and has reduced wartime task order preparation from 16 to 6 hours. Non-programmers can learn how to use it within a 30-minute training period. FRESH at CINCPAC FLEET and TPA at two power plants keep up with real time requirements on a daily basis despite the enormous search spaces they contend with.

The very real and significant benefits of each of these systems are not the only common thread. In addition, the FSKBSs share a number of development, scalability, portability, architec- tural, software engineering, supportability, usability and other obstacles that make them exceedingly difficult to build and maintain. Ted Senator summarized these problems quite aptly as discussant of one of those sessions when he stated "many of the techniques for developing (FSKBSs) are still ad-hoc. Many open issues remain."

Theories for Ontological Engineering: Is it (Knowledge) Soup Yet?

A goal of FSKBSs is to assemble a huge knowledge base in order to overcome the brittle behavior of smaller expert systems when they run into unforeseen problems. One of the differences between building huge KBs and that of building smaller KBs lies in the need to give the machine a layer of insight into the categories of knowledge that are available for a given problem solving effort. The machine could then use this ontology (literally a taxonomy of categories) to decide at what level of detail to try and address a problem, to retrieve instantiations of the ontology, to reason about acquiring further knowledge, and to create expectations for the types of knowledge that the problem solving session should be seeking. Adding this layer to FSKBS is called ontological engineering.

One of the most significant theories presented at the workshop was John Sowa's "Knowledge Soup," which tackles the difficult questions associated with the lack of a universal, consistent ontology. This obstacle often results in FSKBS developers lamenting the "ontological swamp". John's approach suggests an escape route via permitting hopelessly inconsistent chunks (at the ontological, semantic, and representational levels) to co-exist in a "primordal knowledge soup" and by relaxing the constraint for the top levels of the ontology to be stable. The very metaphor of soup, (i.e., nurturing, life-sustaining), captures the effect that John's theory offered to the workshop participants.

Knowledge soup as a theory is significant because it transfers the dilemma from the shoulders of the ontological engineer, who no longer needs to resolve the inconsistencies, to the machine's processing algorithms. The suggestion is for an abductive knowledge association stage where candidate chunks are retrieved from the "soup," followed by a deductive stage where problem sol- ving methods proceed according to canons of formal logic. Prospector II provides an example of this approach.

Prospector II was presented by Dick McCammon in the lessons learned session of the workshops. He explained how Prospector has evolved from its early days of discovery tasks for a single mineral to its current form of supporting multiple tasks for 86 separate mineral models. In this sense it is more than 86-fold larger than the original Prospector. However, Prospector II is an excellent working example of Sowa's theory. The 86 mineral models can be thought of as the soup from which the abductive stage associates candidates relevant to the geological information of the current problem. The deductive stage formally processes those candidates, seeking the best match.

The final presentation under this topic area, given by Ira Monarch and Eswaran Subrahmanian, offered a clever alternative to Sowa's soup that led to much productive discussion about how one might finesse the top level of an ontology (for specific applica- tions). They suggested that in relatively narrow, real world domains one can hope to find consensus that would permit a workable ontology to be constructed, and the inconsistencies to be over- looked. Monarch and Subrahmanian's approach has to be taken on a case by case basis. However, instances were also cited in the discussion of highly contentious organizations where consensus building would obviously not succeed. Their argument is supported by a case study from a real application.

While Monarch and Subrahmanian's paper also contributes to the knowledge server topic area (e.g., it integrates hypertext, knowledge and data) it is relevant to ontology since they show how the ontology provides organizing principles for supporting interactive learning and incrementally automating knowledge acquisition. These Knowledge Server functions are discussed in the next section.

Knowledge Server Research Issues

One of the areas that stood out at the workshop as to where FSKBSs seem to "live or die" is how they handle Knowledge Server issues. A Knowledge Server is analogous to a file server that transparently handles file and message traffic needs, except the Knowledge Server transparently handles all the knowledge manipula-

tion needs. The Knowledge Server is the engineering equivalent of what makes knowledge soup possible. It permits and facilitates the exchange of knowledge between different representations (e.g., rules, objects, cases, and nets) and handles both knowledge and data management operations. It facilitates multi-media (hypertext, graphics, voice, etc.) concerns; it offers innumerable tools for quickly acquiring, discovering, learning and integrating new knowledge into the FSKB; it performs the abductive stage of Sowa's algorithm.

The Knowledge Server term was not widely used during the workshop. However, in the wrap-up session, the participants agreed that it is a suitable metaphor both for explaining what was central to the discussion and for summarizing why so many normally separate communities (expert systems, data base, knowledge acquisition, machine learning, natural language, human-computer cooperation, etc.) felt they belonged in the same workshop.

The consensus was that these communities had to work together if FSKBSs were to be successful. However, there were occasionally strong positions taken about whether all of these communities did or did not have something to contribute to each other. Notably, Ed Feigenbaum stated that at present there is no reason for AI people and database people to meet together in the same room or conference, because progress in KBS is limited by representation issues and not by the performance issues that form the primary focus of database research. Specifically, INTERNIST (CADUCEUS) is a FSKBS that diagnoses 500 diseases and searches through a half a million facts all on a 386 chip with no need for data base technology. Cyc associates across several hundred thousand knowledge chunks without redress to data base technology. The outcome of the discussion was that while AI purists can probably proceed with safety in an isolated mode, the design and construc- tion of practical FSKBS requires use of existing databases.

The papers included under this topic area originally were presented in several different workshop sessions primarily because there is no prevailing theory that properly integrates them. Unlike ontology, which is a theory-rich topic area that requires more practical application to flesh out its engineering problems, the Knowledge Server topic area is theory-poor and engineering- rich. That is, many researchers are developing the Knowledge Server parts bottom-up, on a need-driven basis, and in the absence of an integrating theory or framework. Research is needed on a top down approach to the Knowledge Server concept. However, such research is intimidating due to the vastness of the subject area.

One talk presented at the workshop, by Brian Gaines, was actually the inspiration for the Knowledge Server concept as elaborated here. For example, Gaines stresses that FSKBSs should be considered *primarily* as devices for supporting social processes in organizations and as aids for facilitating the multi-media electronic flow of knowledge soup between communities of experts, users and machine agents. However, his proposal is also ultimately one for a server that can achieve many of the goals delineated at the outset of this section.

One of the Knowledge Server ideas captured by Gaines is that remote nodes might use very different representation and infer- encing mechanisms on shared knowledge. For example, there might be a case-based node, a chaining node, an inductive node, a hyper- media based explanation node, a database node, etc. These topics were a central concern of the workshop participants.

One relief measure proposed for future FSKBSs is the idea that standards could eventually be written which would identify what shell attributes (representation, inferencing, paradigms, etc.) would be desired for each task type. These standards could help researchers compare their needs to what commercial shells offer and thereby identify what might best fit the specific needs of their planned FSKBS. Apparently several professional societies are in the early stages of authoring such standards. Also, DARPA held a Santa Barbara workshop during the Spring of 1990 which addressed related standards questions and came forth with a Knowledge Interface Format (KIF) that could facilitate exchange of knowledge between the diverse Knowledge Server nodes. Ed Feigenbaum explained that as a first step the KIF is now being built by Mike Genesereth and it is being experimented with for knowledge exchanges between three separate centers at Stanford University.

In the plenary session following the workshop, Mike Geneser- eth further highlighted the value of the KIF for facilitating agent oriented programming, where individual agents can exploit diverse shells (e.g., CBR vs. expert systems) and still be able to share knowledge with other agents. He pointed out that the KIF language is intended exclusively for machine to machine interfaces and actual code must be written for FSKBS developers for the specific interfaces they require. Ideas like the KIF respond to Knowledge Server interface programming problems by potentially providing a standard language that can be adopted by FSKBSs throughout the world.

Agent-oriented architectural aspects of the Knowledge Server were of concern to the participants and were directly addressed in two talks. One example presented was an FSKBS development environ- ment called Star, that Lockheed is creating for use in complex realtime systems like their Air Land Battle Management project for DARPA. Star is an exploration of how to deliver utilities, (editors, toolboxes, etc.), shared knowledge representations (objects, geometric displays, maps, messages), and services (windows, inference engines, mapping systems, etc.) in a dis- tributed agent-oriented programming environment. A second example was a presentation on the requirements, issues, and protocols of a distributed blackboard suitable for fault tolerant, realtime, continuous operations needed in the Shuttle Firing Rooms. The goal is to better integrate the operation of a number of individual expert systems currently deployed in advisory roles. Further discussion revealed the relative infancy of the field and the long term nature of such efforts.

Virtually every FSKBS discussed had considerable difficulty with the knowledge to data interfaces and management needs of the Knowledge Server. Three talks gave direct insight into the types of solutions being implemented. Larry Kerschberg stressed the role that expert database systems already are playing in supporting multiple views of the knowledge soup (data, constraints, knowledge, events, cases, etc.). He described extensions of the classical data model that are being made to

include knowledge-based and object-oriented concepts: this is evolving into an Object-Oriented Knowledge Data Model (OOKDM). Finally, he made the interesting point that due to the dynamics of the database element it is now becoming necessary to knowledge engineer the KB for knowledge it doesn't yet hold. Ken Kaufman and Ryszard Michalski presented an environment for managing multiple KBs while handling discovery and learning from existing KBs. They have developed a domain indepen- dent toolbox for applying clustering, concept discovery, and empirical induction that automatically extracts new knowledge from data bases. The process extends the knowledge base and simultaneously manages the KB evolution needs.

The machine learning aspects of the Knowledge Server were further addressed by Larry Hunter, who suggested a new paradigm, "planning to learn". Here the machine interviews the user and jointly forms a plan of what to learn from the data bases, how to learn (i.e., which of about a dozen learning techniques to use), and in which knowledge base (agent, representation, location) to deposit the results. All three talks addressed systems at the robust working prototype stage, and predicted the machine learning, data base, and knowledge update management aspects of the Knowledge Server were on the verge of becoming widely available FSKBS tools in the next few years. A cautionary note was offered by Alberto Segre who expressed considerable doubt that one could scale up the existing batch of machine learning tools to the level of activity required in FSKBSs.

The human expert interviewing aspects of the Knowledge Server were also the subject of much attention, and six FSKBSs with separate knowledge acquisition subsystems were presented. Those of Monarch, Gaines, and Hunter have already been addressed. John Boose presented a significant attempt to automate the corporate or group memory, particularly during large scale design projects and with special support for derivational aspects of tradeoff studies. Barry Silverman in turn suggested that experts are prone to judgmental bias due to cognitive limits on processing large and uncertain problem spaces. He presented an approach in which the interviewing system is also capable of doubting, trapping, and critiquing the expert, so as to improve his inputs before they ripple through the large KB. Mike Freeman was a discussant of both of these talks and pointed out a number of limits. For instance, some aspects of the design knowledge are easier to capture than others, and the approach is missing a taxonomy of tools for acquiring different types of knowledge. He further pointed out the judgment critiquing approach would be difficult in practice, since there are few models of intuition and bias on which to base the system. Finally, Freeman suggested that both efforts need to collect and present additional empirical results for the potential benefits to be better understood.

Ralph Weischedel also presented a knowledge acquisition system that BBN has used to gain an order of magnitude speed up in the deployment of natural language interfaces to large scale databases. In a second talk, he went on to elaborate that natural language technology has a number of successful applications in data base interfaces, and it is now ripe for exploitation in other similarly narrow domains. As an example of future directions, he cited an interface BBN has developed to the FRESH and CASES systems, as well as to the underlying databases and analytical support and display systems. He predicted that in the next five years there will be a number of FSKBSs and other types of systems utilizing natural language interfaces, and that by the end of that half decade another order of magnitude will be attained in the speed of deploying such interfaces.

The issues of developing huge, heterogeneous knowledge bases that amplify group activity are now sufficiently clear that researchers are directly addressing them. A half decade from now a number of the missing implementation, scale-up, portability, standardization, performance, and empirical evaluation details will be available. By that time the Knowledge Server will very likely be a reality and FSKBSs of the future will be built in a much easier fashion. Until that time, however, it is important to benefit from the lessons learned in actual FSKBS efforts attempted to date.

Organization-Computer Collaboration

This issue received no direct discussion during the workshop, but is included here since virtually every working FSKBS presented had considerable effort invested in designing their systems to serve as organization or group activity amplifiers. As a few examples, consider the NASA firing room's use of Heard's advisory agents, the Pacific fleet command center's integration of FRESH and CASES, and Boeing's design team interactions with AQUINAS. In each case the FSKBS must be tuned in to and serve the communication and interactive processes of the group.

These are examples of why Gaines has said FSKBSs should be thought of as social processes. These systems do not have a single man-machine interface but rather have numerous stations that must facilitate not only machine-to-machine and human-machine communication, but also must keep abreast of and support human-to-human problem solving. The presentations made during this session gave a clearer view of the effort that must be expended in designing organization-computer collaborative systems.

A framework for thinking about this problem was included in Silverman's presentation. He asserted that the FSKBS is a vehicle for facilitating organizational communication and collaboration. He also suggested there are many variables and factors that must be researched if successful organizational collaboration is to be guaranteed in FSKBS efforts: e.g., organizational, environmental, task and interpersonal factors must be quantified and factored into a given FSKBS. He presented a model of human-to-human collaboration along with one for human-machine collaboration. Despite these tentative models, there is a dearth of theory that can be exploited for designing FSKBSs.

A good example of a system making a trial and error contribution to this topic area is the Federal Bureau of Investigation's counter-terrorism FSKBS nicknamed "Big Floyd." While this system is notable because it has 40,000 Horn clauses atop a dynamic database with millions of records, its human interface is even more significant. Gary Gardner, the domain expert who guided the system's

development, walked the audience through a large collection of direct manipulation interface screens full of metaphors right out of the average field investigator's work. These screens were highly understandable in terms of keeping abreast of the updates to the terrorist sitings, case by case developments, and rule-induced case conclusions and recommendations. The system currently has several thousand field users and serves as an electronic way for the entire organization to keep in synch with each other as cases proceed. This is a splendid example of where collaboration theory research could benefit from study of actual FSKBS experience.

A second example is the Program Managers Support System (PMSS) which is a collection of 100 modules guided by an expert system tailored for executives. PMSS is specifically intended to keep each member of a program office coordinated with events/decisions affecting the other parts. For example, budget cuts entered into the financial module by the financial expert automatically ripple through to the manpower planning, technical schedule, and logistics planning modules, with appropriate coloration and messages so the experts in those sub-domains can assess and cope with the impacts. Similarly, schedule slippages entered by a single mode expert ripple back to the other modules and experts, and so on. This approach offers organizational coordination enhancement potential for program offices and is currently in use at 19,000 sites in the Department of Defense. Work is needed to examine how systems such as PMSS (or the FBI's) both augment and hinder human collaboration and group productivity. FSKBS designs need to be more predictable in their potential impacts.

National FSKBS Initiatives

Despite the many obstacles to advancing the state of the art in FSKBS technology (if not theory), combined with the shortcomings of systems based on narrow KBs, a number of "national initiatives" have recently been suggested. The size of these initiatives, with regard to both funding levels and number of organizations involved, indicate a major committment to dramatically improving the productivity of the countries, industries, and fields in which they apply.

The workshop was fortunate to have presentations on three such endeavors. In addition, Ed Feigenbaum presented much of the background motivation for MCC's Cyc Project and other national initiatives. The common goal of these efforts is to hand code very large KBs.

Dr. Feigenbaum's fundamental point was that his "knowledge is power" principle was arrived at in 1968 as the first empirically derived principle of AI. All the expert systems built since then are further experimental verifications of that principle. It is time, he maintained, to launch the second generation of KBSs that exploit this principle at levels never before attempted — i.e., with millions, tens of millions, and greater numbers of knowledge chunks. Since expert systems with thousands of knowledge chunks have led to significant productivity gains and other benefits, he felt corresponding scale-up in benefits from truly massive KBs should also be expected.

Tests of this premise are expensive and there are few organizations able to get on with the task of actually assembling such a system. Doug Lenat and MCC have taken the first step, and others are now following suit. By the mid 1990's there will be several such systems, and these will eventually become national assets. A few more decades of further developments along these lines will lead to major breakthroughs in the general intelligence and world-awareness of the systems we routinely build.

The first national initiative presentation was by Yokoi and Uchida of the Japanese Electronic Dictionary Research (EDR) Institute. Japan is investing 14 billion yen (over 100 million dollars) in the ten-year development of a Japanese-English dictionary and translation system. As part of the 5th generation computing project, Japan invested heavily in a grammar parsing natural language approach. After several years the system could parse quite well but had no vocabulary with which to speak. The EDR project was launched in 1986 as a joint government-industry response to this problem.

The EDR dictionary is an electronic lexical knowledge base emcompassing a 250,000 word dictionary, and a 100,000 concept dictionary in Japanese, plus equivalently-sized dictionaries in English. All words and concepts are cross-linked within and between each language via a co-occurrence dictionary and a bi-lingual dictionary. Initially the EDR dictionary was based on Schankian conceptual dependencies and still makes use of relations and pseudo-relations to establish many of the linkages. However, the design appears to have migrated to Cyc-like nets: the precise theoretical basis for the approach was unclear.

Half-completed by 1989 and due to be finished in 1996 or 1997, the EDR project employs approximately 260 "knowledge workers". The team consists of a few highly competent knowledge engineers who are designing the dictionary and deciding on the linkages. The remainder of the knowledge workers are primarily "ordinary people", including little old ladies from fishing villages, employed at low cost to both encode the knowledge and to be sure it is understandable at a lay person's level.

The second national initiative presentation by Dietmar Roesner was preceeded by an introduction by Alexa T. McCray from the National Institutes of Health. Roesner reported on recent proposals submitted to West German and other European funding agencies. The goal of the proposals is the establishment of a center, similar to the Japanese EDR, for the development of a lexical knowledge base for NLP applications. It would also support the coordination of lexical researchers located in a variety of institutions, including corporate R & D centers, universities, and publishing houses. Roesner and McCray both pointed out that the development of large scale lexical databases is essential to the success of NLP applications. These applications include machine translation, natural language front-ends, speech processing and spelling, style and grammar checkers, as well as the NLP aspects of information retrieval systems, computer assisted instruction, office automation, networking and telecommunications, and electronic publishing. The recognition that the potential applications of NLP are so varied has led the proposers, under the leadership of the German Institute for Computational Linguistics (Institut fuer Maschinelle Sprachverarbeitung), to develop a set of desiderata for a large scale lexical knowl-

edge base. These include content issues, software and hardware issues, and inter-disciplinary issues. The lexical knowledge base must be "polytheoretic ... the idea is to provide a common superset of the needs of a variety of these [recent linguistic] theories and to take this as a basis for the lexicon to be created." In addition, the knowledge base should be multilingual. McCray noted that in the European setting the multilingual aspect cannot be ignored. The lexical knowledge base must be implemented in a software environment that allows ease of creation, maintenance and updating. Finally, the effort must be inter-disciplinary, since it potentially involves expertise in not only linguistics, but also software engineering, knowledge representation, communications technology, database storage and retrieval and copyright issues.

The first two national initiatives concern the development of lexical knowledge bases for large scale natural language processing. Both include a significant investment of time and effort. This raises the question of such projects pursuing automated knowledge acquisition approaches such as presented by Weischedel. However, automated approaches apparently are being neglected in these and other initiatives at this time. The feeling is that no shortcuts to hand coding these KBs is possible.

The third and final national initiative, like the European one, is also in the proposal stage. Unlike the other two, however, it is not for a natural language system, but is instead for a massive FSKBS in the area of science and engineering knowledge. The entire final 2 hour session of the workshop was devoted to this proposal including presentations from Steve Cross of DARPA, Edward Feigenbaum from Stanford University, and Chandrasekaran from Ohio State University.

The goal of the project is to develop a national encyclopedic resource comparable to an online machine usable set of books containing sharable knowledge in the fields of electromechanical devices. This knowledge would be used by the machine to reason about the world of engineering and design, and to help users in their efforts to build simulators, to formulate models, and to diagnose and explain device behavior. Anyone who works with the science or engineering of the encompassed devices could benefit from the machine already understanding their domain. When constructing knowledge bases for specific applications one would, in effect, merely extend the encyclopedia.

Some of the many benefits of such an encyclopedia are: (1) the system would be capable of reasoning broadly about the domain and would not reach the abrupt "cliffs" of the current generation of narrow KBs; (2) adding knowledge to the encyclopedia would be easier than traditional knowledge engineering since the machine would already understand the domain and could readily interview (and possibly critique) the user for specialized inputs; (3) the encyclopedia would retain any and all additions to it for re-use and sharing among future users; (4) this could serve as a useful vehicle for researching and developing all of the ontology, knowledge soup, Knowledge Server, standards and organization-cooperation ideas mentioned in the workshop; (5) approaches and technology developed for this engineering domain could be re-used to create encyclopedias for other domains.

In addition to the three presenters, a number of AI scientists have apparently been regularly interacting and developing the concept for this engineering FSKBS including S. Addanki at IBM, R. Davis at MIT, K. Forbus at U. of Illinois, J. Dekleer at Xerox PARC and D. Lenat at MCC. The long term goal would be to involve hundreds of groups from industry, professional societies, and elsewhere to evolve the shared ontologies, to develop parts of the encyclopedia, to test the knowledge content ("Can you re-use what I've put in?"), to develop the needed standards, etc.

Several technical details were presented by Chandrasekaran concerning generic tasks as well as the need to represent both conceptual (symbolic) and perceptual (pictorial) cases in a domain. He argued the human mind often reasons about designs pictorially (e.g., a broken gear tooth's effect on an interconnecting gear) and he indicated new knowledge representation, indexing and reasoning processes are being investigated by his students to facilitate this kind of visual engineering thinking in the FSKBS.

The discussion from the participants raised concerns about the vastness, Knowledge Server (database aspects in particular), and knowledge sharability aspects of the proposed FSKBS, but it was generally agreed that such a project would serve as a watershed capable of opening the doors to an entire new generation of expert systems capability, benefits and impacts.

Concluding Comments

Full-sized knowledge based systems are one of the major research frontiers in the AI field. By tackling national initiatives for hand-coded knowledge bases with a million or more knowledge chunks, or by directly researching and developing the ontology and component knowledge server theories and technologies, researchers are paving the way for major breakthroughs in the broad, general intelligence of systems we routinely build. Future expert systems, rather than being trained on narrow, brittle views of the domain, will increasingly be schooled in the underlying models, principles and knowledge of complete fields of study. While the vast majority of expert systems fielded annually will continue to be in the small- to medium-sized category for the foreseeable future, extremely large FSKBSs should begin to appear in the 1990s in specific disciplines such as natural language, common sense reasoning, and engineering. Like hardware generation advances, these FSKBSs will represent the second generation of capability, benefits, and intelligence of the expert systems field. Like hardware, many of these FSKBSs will be reusable for multiple application projects.

To reach this next generation of "shell" technology, the skill mixes of our development teams will need to be dramatically altered. Rather than using a few computer scientists and a marketeer, the second generation of expert systems technology will need to be developed by a team with as many specialties as included in the attendance at this workshop. Specialists will be hired for ontological engineering, knowledge engineers will have to organize and "tag" the domain knowledge according to the ontology, and large numbers of "ordinary people" will be paid to encode it into the shell's background knowledge base. On the knowledge server side, the shell development team will

need to include members from a variety of communities such as, but not necessarily limited to, production systems or logic programming (engines and representations), data model building, machine learning, knowledge acquisition, natural language, hypermedia, modeling and simulation, and distributed software systems engineering. Finally, there will also need to be a team of behavioral practitioners responsible for designing the organizational-computer collaboration features such as direct manipulation metaphors, tutoring capabilities, group productivity enhancement techniques and knowledge sharing/reuse concepts.

But just hiring such team members alone will not assure the future of the FSKBS field. There are major gaps in our understanding the theory of ontologies, knowledge servers, collaboration, and knowledge reuse approaches. Researchers need not only to pioneer new theories for these and related FSKBS topics, but also to address the scale-up issues associated with applying the resulting techniques to large-scale KB problems. Researchers also need to evaluate current FSKBS applications to extract lessons learned and to compile design-related insights. Unfortunately, few researchers are prepared to think as broadly as or to work for the durations required to fill these FSKBS needs.

The costs and scale of effort of researching and developing FSKBS shells is intimidating. The potential benefits, however, seem even more dramatic. FSKBS shells portend the coming of the "knowledge age", an era where machines begin to truly understand our world and to collaborate with us as equals. The founding vision for the AI field is of machines that do our bidding without needing to be programmed. The advent of FSKBS shells will be a significant step in that direction.

While all at the workshop shared the common goal of reducing brittleness through the creation of larger scaled KBs, it seems safe to conclude two major paths are being taken toward that goal. On the one hand the advocates of the knowledge principle are pioneering in the areas of ontology development and national initiatives to launch man-centuries of KB coding effort. They are doing this in the belief that there are no short cuts to the creation of very large KBs. On the other hand, there are the application builders, knowledge server developers, and collaborative systems researchers who believe that FSKBSs must ultimately be multifaceted. It is worthwhile to attempt to advance and integrate a number of technologies that eventually will be needed by FSKBSs: e.g., knowledge acquisition, machine learning, natural language, database, and distributed agent frameworks to mention a few.

In the final analysis both approaches are necessary and complementary. No nation has the resources to continue large scale KB encoding by hand on an indefinate basis, and even if they did, hand coding would never keep up with the realtime problem solving, dynamic data flow, and collaboration requirements of current day FSKBSs. Similarly, the relative immaturity of current knowledge server technologies render most FSKBS developers to the position of crafting what they can as the need evolves. The concensus appears to be that in one to two decades, progress on both sides — manual knowledge coding and scale-up of knowledge servers — may both mature to the point where their merger could lead to an intelligent system that is both broad and able to solve real world problems. Until then, the two paths seem destined to be walked at first separately, and then increasingly in tandem.

Acknowledgements

The authors gratefully acknowledge the efforts of the workshop sponsors and attendees, and especially R. Gregory Wenig, workshop Co-chair, and Quasir Durrani, who helped with the organization and correspondence.

About the Authors

Barry G. Silverman is Director of the Institute for AI at the George Washington University and is author/editor of over 12 books or proceedings some of which include: *Expert Systems for Business, Military Applications of AI, AI Systems in Government,* and *Building Expert Critics* (pending).

Arthur J. Murray is Manager of Advanced Technology at McDonnell-Douglas in McLean, Virginia. He received his Ph.D. in engineering administration from the George Washington University in 1989.

1990 AAAI Spring Symposium Report:
Theory and Application of Minimal-Length Encoding

Edwin P. D. Pednault

This symposium was very successful and was perhaps the most unusual of the spring symposia this year. It brought together for the first time distinguished researchers from many diverse disciplines to discuss and share results on a particular topic of mutual interest. The disciplines included machine learning, computational learning theory, computer vision, pattern recognition, perceptual psychology, statistics, information theory, theoretical computer science, and molecular biology, with the involvement of the latter group having lead to a joint session with the AI and Molecular Biology symposium. The unifying topic was the problem of finding the shortest encoding of a body of information in a language rich enough to admit many possible encodings.

This particular problem arises in different forms and for different reasons in each of the disciplines mentioned above. In machine learning and computational learning theory, minimal encoding can be used to infer a general theory from a set of observations. If one employs a language for encoding observations in terms of general rules or properties, the shortest description will be an encoding of the desired theory together with a minimal set of additional information necessary to derive the observations from the theory.

In computer vision and pattern recognition, scene analysis can be equated with finding the shortest encoding of an image in a language for describing images in terms of objects and their visual properties. In perceptual psychology, this cod-