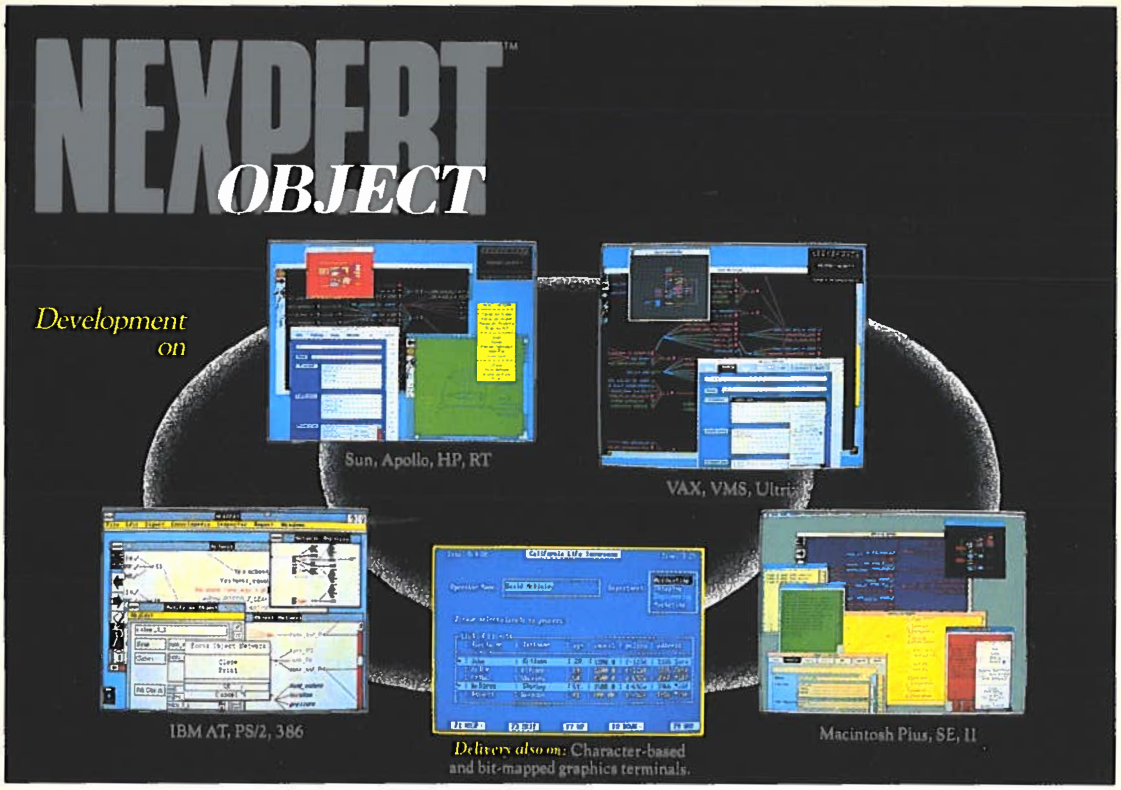
Canadian Artificial Intelligence
Intelligence Artificielle au Canada

January/janvier 1990 **No. 22**

An official publication of CSCSI, the Canadian Society for Computational Studies of Intelligence
Une publication officielle de la SCEIO, la Société canadienne pour l'étude de l'intelligence par ordinateur



REASONS FOR NEXPERT'S SUCCESS

Links to popular conventional languages

– C	– FORTRAN	– Ada
– COBOL	– Pascal	– Assembly

Effective linkage with other databases and to own database – provides direct system in and out calls through the Library

– Oracle	– Sybase	– Ingres
– Informix	– RdB	– SQL/DS
– dBase	– Lotus 1-2-3	– Excel
– Guide	– Hypercard (Q1/89)	

NEXTRA – first truly useful Knowledge Acquisition aid

- graphically induced acquisition process with automated repertory grids
- hierarchical clustering with spatial representations
- inductive engine for example-based analysis
- graphical feedback of elicited knowledge

Linkage to other graphics packages
Ability to graphically represent macrostructure of the knowledge base

– DataViews	– AiVision	– MACPAINT
– Ease+	– PCPaint/Brush	

Cross-compatibility across a wide range of platforms – Quick and efficient adaptation to new platforms due to advanced Software Engineering techniques.

- development on VAXSTATION II, III & 2000 (VAX/VMS, ULTRIX) with delivery on all VAX computers
- Sun, HP, Apollo, and MicroVax workstations
- Hewlett-Packard 9000 series 300 and 800
- 80386 Unix workstations running X-windows
- IBM PC AT, PS/2 and compatible 80386-based PCs running Microsoft Windows
- Macintosh family (Plus, SE, II, IIx)
- IBM mainframe under VM (the 30xx, 43xx, and 9370 machines)
- Several foreign platforms (European, Japanese)

Runtime Package

- serious consideration given to runtime/delivery issues
- dual user interface concept: *end-user interface* not identical to *developer interface*
- highly automated end-user interface design tool

NEXPERT is a registered trademark of Neuron Data.
Applied AI Systems, Inc. is an authorized Canadian dealer.

Phone or write for more information

AI

Applied AI Systems, Inc.
Gateway Business Park
300 March Road, Suite 602
Kanata (Ottawa), Ontario, Canada K2K 2E2
Tel. (613) 592-3030 • Fax. (613) 592-2333

Canadian Artificial Intelligence

Intelligence Artificielle au Canada

No. 22 January/janvier 1990

Contents

Canadian Society for Computational Studies of Intelligence

Founded in 1973

CSCSI is the Canadian society for the promotion of interest and activity in artificial intelligence. It conducts workshops and fully refereed national conferences, publishes this magazine, sponsors the journal Computational Intelligence, and coordinates activities with related societies, government, and industry. To join CSCSI, use the membership form in this issue. Non-Canadian members are welcomed. CSCSI is affiliated with the Canadian Information Processing Society and International Joint Conferences on Artificial Intelligence, Inc.

Memberships in CSCSI:
Membership form is on the last page. Please send subscriptions, memberships, and changes of address to:
CSCSI, c/o CIPS, 243 College Street, 5th floor
Toronto, Ontario, CANADA M5T 2Y1

Société canadienne pour l'étude de l'intelligence par ordinateur

Fondée en 1973

SCEIO est la Société canadienne encourageant l'intérêt et la recherche en intelligence artificielle. Elle organise des ateliers ainsi que des conférences nationales avec évaluation des articles soumis. Elle publie ce magazine, subventionne le journal Intelligence Informatique, et coordonne toute interaction avec des sociétés parallèles, le gouvernement, et l'industrie. Pour devenir membre de la SCEIO, veuillez utiliser le formulaire d'inscription de ce numéro. Les non-canadiens sont bienvenus. La SCEIO est affiliée à l'Association canadienne informatique, et aux International Joint Conferences on Artificial Intelligence, Inc.

Cotisations dans la SCEIO :
Le formulaire d'inscription est à la dernière page. Prière d'envoyer tout abonnement, cotisation, et changement d'addresse à:
SCEIO, c/o CIPS, 243 College Street, 5th floor
Toronto, Ontario, CANADA M5T 2Y1

President/Président: Dick Peacocke, Bell-Northern Research, Box 3511, Station C, Ottawa, Ont K1Y 4H7.
613-765-2629. BITNET: richard @bnr.ca
Past-President / Président Précédent: Gordon McCalla, Dept. of Comp. Sc., U. of Sask., Saskatoon, Sask S7N 0W0.
306-966-4902. BITNET: aries@sask
Vice-President/Vice-Président: Renato De Mori, School of Computer Science, McGill U., Montréal, Qué H3A 2K6.
514-398-7072. UUCP: renato@musocs
Secretary/Secrétaire: Bill Havens, Expert Systems Lab, Simon Fraser University, Burnaby, B. C., V5A 1S6, havens@cs.sfu.ca
Treasurer/Trésorier: Jan Mulder, Dept. of Math, Stats and Comp. Sc., Dalhousie U., Halifax, NS B3H 3J5.
902-424-3356. CDNnet: mulder@cs.dal.cdn

Canadian Artificial Intelligence

Intelligence Artificielle au Canada

Founded in 1974 as / Fondée en 1974 en tant que CSCSI/SCEIO Newsletter

Submissions:
Canadian Artificial Intelligence is published quarterly by CSCSI/SCEIO and is a benefit of membership in the society. *Canadian AI* solicits contributions in English or French on any matter related to artificial intelligence, including: articles of general interest; descriptions of current research and courses; reports of recent conferences and workshops; announcements of forthcoming activities; calls for papers; book reviews and books for review; announcements of new AI companies and products; opinions, counterpoints, polemic, controversy; abstracts of recent publications, theses, and technical reports; humour, cartoons, artwork; advertisements (rates upon request); anything else concerned with AI. Paper or electronic submissions are welcome. Electronic submissions are preferred and should be unformatted. Canadian AI is published in January, April, July, and October. Material for publication is due six weeks before the start of the month of publication.

Advertising:
Advertising rates and press kits are available upon request from the address below, or by phoning 403-297-2600.

Contributions :
L'Intelligence artificielle au Canada est publiée trimestriellement par la CSCSI/SCEIO, et est offerte gratuitement aux membres. *L'IA au Canada* encourage les contributions, en français ou en anglais, portant sur l'intelligence artificielle. Ceci comprend: des articles d'intérêt général; des descriptions de recherche courante et de cours; des rapports de conférences récentes et d'ateliers; l'annonce d'activités à venir, et des requêtes d'articles; des critiques de livres ainsi que des livres à critiquer; l'annonce de nouvelles compagnies en IA et de leurs produits; des opinions, des répliques, tout ce qui est polémique; des résumés de publication récentes, de thèses et de rapports; des trucs humoristiques ou artistiques, de bandes dessinées; des annonces (s'enquérir des frais); tout autre matériel touchant à l'IA. Contributions, sur papier ou par courrier électronique, sont bienvenues. Nous préférons le courrier électronique mais les soumissions ne doivent pas avoir un format. L'IA au Canada apparait en janvier, en avril, en juillet, et en octobre. Toute communication à publier doit nous parvenir au moins six semaines avant le début du mois de parution.

Réclame:
Les prix pour les annonces et les trousses pour la presse sont disponibles sur Écrivez à l'addresse à la gauche ou téléphonez 403-297-2600.

Please send submissions to / Prière d'envoyer contributions à:
CDNnet: CSCSI@noah.arc.cdn
UUCP: cscsi%noah.arc.cdn@alberta.uucp
or to / ou à: Marlene Jones
Canadian Artificial Intelligence
Alberta Research Council
6815 8th Street NE, 3rd floor
Calgary, Alberta, CANADA T2E 7H7

Book reviews and candidate books for review should be sent to:
Envoyez des critiques de livres ainsi que des livres à critiquer à:
CDNnet: gh@ai.toronto.cdn
CSNET: gh@ai.toronto.edu
UUCP: gh@utai.uucp
or to / ou à: Graeme Hirst, *Canadian Artificial Intelligence*
Department of Computer Science, University of Toronto
Toronto, Ontario, CANADA M5S 1A4

Editor/Editeur: Roy Masrani
Editor Emeritus/Rédacteur emeritus: Graeme Hirst
Translation/Traduction: Anne Parent, Diane Lefebvre, Raymond Aubin
Production: Marco Ariano, Janet McCracken
Advertising/Réclame: Marco Ariano
Publications/Publications: Graeme Hirst

Canadian Artificial Intelligence is published quarterly by the Canadian Society for Computational Studies of Intelligence (CSCSI).
Intelligence Artificielle au Canada est publiée trimestriellement par la Société canadienne pour l'étude de l'intelligence par ordinateur (SCEIO).
Second Class Mail Registration No. 7373

ISSN 0823-9339

Executive Notes

Membership of the NRC Associate Committee on Artificial Intelligence

Membres du CNR comite associe sur l'intelligence artificelle

1989-1990

Mr. L. Bolger
Vice-President, Research, Shell Canada Ltd., P.O. Box 100, Station M, CALGARY, Alberta T2P 2H5
Tel: (403) 232-4800 Fax: (403) 232-4840

Dr. Marie-Michele Boulet
Departement d'Informatique, Universite Laval, Pavillon Pouliot, Cite Universitaire, QUEBEC, P.Q. G1K 7P4
Tel: (418) 656-7979 Fax: (418) 656-7063 Netnorth: BOULET@LAVALVM1

Dr. N. (Nick) Cercone
Director, Centre for Systems Science, Simon Fraser University, BURNABY, B.C. V5A 1S6
Tel: (604) 291-3229 Fax: (604) 291-4045 Nick@cs.sfu.ca

Dr. E. Chang
Alberta Research Council, 3rd Floor, Digital Building, 6815-8th St., N.E., CALGARY, Alberta T2E 7H7
Tel: (403) 297-2600 Fax: (403) 275-3003

Dr. R. De Mori – Chairman
Centre de recherche informatique, de Montreal (CRIM), 1550, boul. de Maisonneuve ouest, MONTREAL, P.Q. H3G 1N2
Tel: (514) 398-7072 Fax: (514) 398-3883 - McGill
Tel: (514) 848-3990 Fax: (514) 848-8892 - CRIM

Mr. P. George
Director of Engineering, Systems Division, MacDonald, Dettwiler & Assoc., 13800 Commerce Parkway, RICHMOND, B.C. V6V 2J3
Tel: (604) 278-3411 Fax: (604) 278-1285

Mr. K. Hayes
Policy and Planning, Research and Legislative Department, Canadian Labour Congress, 2841 Riverside Drive, OTTAWA, Ontario K1V 8X7
Tel: (613) 521-3400 Fax: (613) 521-4655

Dr. E.G. Manning
Professor & Dean, Faculty of Engineering, University of Victoria, VICTORIA, B.C. V8W 2Y2
Tel: (604) 721-8611 Fax: (604) 721-8653

M. R. Manoliu
Project Manager, Task Force on Expert Systems, Developpement technologique, Hydro Quebec, 17e etage
455 ouest, boul. Rene Levesque, Montreal, que. H2Z 1Z2
Tel: (514) 289-3877 Fax: (514) 874-0810

Dr. G. McCalla
Dept. of Computational Science, University of Saskatchewan, SASKATOON, Saskatchewan S7N 0W0
Tel: (306) 966-4902 Fax: (306) 966-8710
Bitnet: Aries@sask.usask

Dr. G. (Gordon) MacNabb
Precarn Associates Inc., 30 Colonnade Road, Suite 300, NEPEAN, Ont. K2E 7J6
Tel: (613) 727-9576 Fax: (613) 727-5672
Irapnet: GAVEL@A1.ATOTT2.NRC.CA

Dr. J. (Jan) Mulder
Dept. of Mathematics, Statistics and Computer Science, Dalhousie University, HALIFAX, N.S. B3H 3J5
Tel: (902) 424-3356 CDNNET: mulders@cs.dal.cdn

Dr. R. Peacocke
Manager of Knowledge Technology, Bell Northern Research, Gateway Bldg., P.O. Box 3511, Station C, OTTAWA, Ontario K1Y 4H7
Tel: (613) 765-2629 Fax: (613) 763-4222
Bitnet: Richard@bnr.ca

Prof. Z. Pylyshyn
Director, Centre for Cognitive Science, Department of Psychology, Social Science Centre, Room 7332, The University of Western Ontario, LONDON, Ontario N6A 5C2
Tel: (519) 661-2063 Fax: (519) 661-3292

Mr. J. Scrimgeour – Secretary
Senior Advisor, National Research Council, Building M-50, Room 205, OTTAWA, Ontario K1A 0R8
Tel: (613) 993-9000 Fax: (613) 952-7998

Mr. George F. Sekely
Vice-President, Computers and Communications, Canadian Pacific, 40 University Avenue, Suite 1200, Toronto, Ontario M5J 1T1
Tel: (416) 863-8216 Fax: (416) 823-8260
MERLIN: SEK 0001

Dr. B.J. Smith
Acquired Intelligence, #205 - 2951 Tillicum Rd., VICTORIA, B.C. V9A 2A6
Tel: (604) 383-8935 Fax: (604) 383-2835

Mr. G. Thomas
Partner, Price Waterhouse, Suite 1100, 180 Elgin Street, OTTAWA, Ont. K2P 2K3
Tel: (613) 238-8200 Fax: (613) 238-4798

Regular Observers/Observateurs

Mr. A. DLubois
Acting - Director, Technology Application Directorate, Information Technologies Industry Br., 9th Floor East, ITIB, Industry, Science & Technology Canada, 235 Queen Street, OTTAWA, Ontario, K1A 0H5
Tel: (613) 954-3452 Fax: (613) 954-8419

Dr. David G. Goodenough
Chief Research Scientist, Head, Methodology Section, Energy, Mines and Resources, Major Projects Office, Canada Centre for Remote Sensing, 1547 Merivale Road, 5th Floor, OTTAWA, Ont. K1A 0Y7
Tel: (613) 952-2760 - office (613) 726-6904 - home
Fax: (613) 952-9783 Email: DGG@CCRS.CDN

Mr. S.A. Mayman
Director, Division of Electrical Engineering, National Research Council, Building M-50, Room 243, Montreal Road, OTTAWA, Ontario K1A 0R8
Tel: (613) 993-9010 Fax: (613) 952-7998

G. Neal – Working Secretary
National Research Council, Building M-50, Room 208, OTTAWA, Ontario K1A 0R6
Tel: (613) 993-3478 Fax: (613) 952-7998

Mr. O. Silverman
Director, Strategic Technologies, Strategic Technologies Planning Div., Industry, Science & Technology Canada, 240 Spark Street, 8th Floor, West, OTTAWA, Ontario K1A 1A1
Tel: (613) 993-7599 Fax: (613) 996-7887

Mr. Vincent Taylor
National Defence Headquarters, MGen George R. Pearkes Building, 101 Colonel By Drive, OTTAWA, Ontario K1A 0K2
ATTN: DRDCS-2
Tel: (613) 995-8008 Fax: (613) 996-0038
Intermet: vktaylor@ncs.dnd.ca

Dr. V. Thomson
Division of Mechanical Eng., National Research Council, Building M-3, Montreal Road, OTTAWA, Ontario K1A 0R6
Tel: (613) 993-9461 Fax: (613) 952-7686

Ms. J. Walden
Director, Strategic Grants Program, Natural Sciences and Engineering Research Council of Canada (NSERC), Centennial Towers, 200 Kent Street, OTTAWA, Ontario K1A 1H5
Tel: (613) 996-2717 Fax: (613) 992-5337

Letters to the Editors

Dear Editor,

I'm writing this letter to complain about the way CIPS has handled my membership in CSCSI and my subscription to "Computational Intelligence". I've been a member of CSCSI since 1984 and I've been subscribing to Computational Intelligence for almost as long.

Every year, CIPS has managed to bungle either the membership, the subscription or both. I've had to call them every year to straighten things out.

In winter 1989, I received a notice from CIPS that my membership and subscription were going to run out soon. Since this was the first such notice I ever received and it was correct, I almost wrote them a letter of congratulations.

But, the good service was short lived. In August of 1989 I called in an address change. Well, that was the end of my membership and subscription. Instead of changing the address, they cancelled everything.

Why is CSCSI Associated with CIPS? I have serious doubts about a society for computer professionals that can't even handle a simple program for keeping track of members. If CSCSI is only associated with CIPS so that we can have our membership dues collected by them, then I propose that we affiliate ourselves with the Columbia House Record Club instead! This company probably has some of the best membership software in the world.

Yours sincerely,

Andre Trudel
Jodrey School of Computer Science
Acadia University

National Computer Network To Link Researchers across Canada

OTTAWA (Nov. 23, 1989)- - The National Research Council has selected a proposal put forward by the University of Toronto in association with IBM Canada and Integrated Network Services Inc. (INSINC) to develop a nation-wide computer communications network for Canadian researchers.

As a high-speed backbone network, the national research network (NRNet) will integrate provincial research networks in British Columbia, Ontario, Quebec, and Nova Scotia, for example, and will incorporate future regional networks as they develop. NRNet will also be connected with major research networks in the United States.

NRC, which initiated the estalishment of the network, will contribute $2 million in start-up funding over the next three years.

It is anticipated that the regional networks will contribute another $1.5 million over the same period. Additionally, the University of Toronto will be providing staff time worth about $300,000 and INSINC will provide discounts on the communication lines.

IBM Canada Ltd. will contribute more than $230,000 of hardware and operating system software integral to the development of the system. In addition, the same sophisticated router and network management software used by NSF Net will be contributed by IBM. NSFNET is the National Science Foundation's network linking regional universities and research centres across the United States.

NRNet is expected to be operational in 1990. Through the network, researchers in universities and industrial and government laboratories will be able to share information as well as facilities such as supercomputers, databases, and programs.

"It's essential that Canada pool its scientific expertise, at a time when technological advances depend on a multidisciplinary approach to researh," says Earl Dudgeon, NRC's Vice-President of Engineering. "The network will allow Canadian scientists and engineers in different cities to work together in ways not currently possible."

The development of national computer communications is part of the government's initiative to enhance the effectiveness of the Canadian research community. The NRNet concept was developed by NRC in close collaboration with Industry, Science and Technology Canada (ISTC) and the Department of Communications

Conseil national de recherches Canada Liaison Informatique Nationale Pour Les Chercheurs Canadiens

OTTAWA (le 23 novembre 1989) - - Le Conseil national de recherches a choisi une offre soumise par l'Universite de Toronto, de concert avec IBM Canada et Integrated Network Services Inc. (INSINC), pour la mise au point d'un reseau telematique national destine aux chercheurs canadiens.

En tant que reseau de transmission de donnees a haute vitesse, le national research network (NRNet) integrera des reseaux de recherche provinciaux de la Colombie-Britannique, de l'Ontario, du Quebec et de la Nouvelle-Ecosse, pour ne citer que ceux-la, et, plus tard, incorporera d'autres reseaux regionaux a mesure de leur formation. NRNet sera egalement relie a d'importants reseaux de recherche des Etats-Unis.

La quote-part du CNRC, qui est a l'origine de cette initiative, sera de 2 millions de dollars en capitaux de lancement qu'il versera au cours des trois prochaines annees.

On attend un autre apport de 1.5 million de dollars des reseaux regionaux au cours de la meme periode. D'autre part, l'Universite de Toronto fournira l'equivalent de 300,000 dollars en vacations de personnel, et INSINC offrira des remises sur les lignes de transmissions.

La compagnie IBM Canada Ltd., quant a elle, prevoit d'apporter une contribution de 230,000 dolllars sous forme de materiel et de logicels d'exploitation indispensables au developpement du systeme. En complement, IBM fournira les memes logiciels perfectionnes de gestion de reseau et d'acheminement que ceux utilises par NSFNet. NSFNet est le reseau de la National Science Foundation qui relie entre eux les universites et les centres de recherche regionaux americains.

On prevoit que NRNet sera mis en service en 1990. Grace au reseau, les chercheurs des laboratoires universitaires, industriels et governementaux seront en mesure d'echanger de l'information, et de mettre en Commun du material et des outils comme, par exemple, des superodinateurs, des bases de donnees et des programmes.

"Il est essentiel que le Canada mette en commun son expertise scientifique parce que nous sommes entres dans une ere ou les progres scientifiques sont tributaires d'une approche pluridisiplinaire de la recherche" a declare Earl Dudgeon, vice-president (Genie) du CNRC. "Le reseau permettra aux scientifiques et ingenieurs canadiens en poste dans les differentes regions du pays d'uinir leurs efforts grace a des moyens actuellement hors de leur portee", a ajoute M. Dudgeon pour conclure.

La mise au point d'un reseau teleinformatique national entre dans le cadre d'une initiative gouvernementale visant a accroitre l'effecacite de la communaute scientifique canadienne. Le concept NRNet a ete mis au point par le CNRC en collaboration etroite avec Industrie, Sciences et Technologie Canada (ISTC) et le

(DOC). There has also been extensive consultation with representatives of the user community.

The same three federal organizations are also collaborating on a study, sponsored by ISTC, to determine the feasibility of establishing a national broadband multimedia network to enhance collaborative research in strategic technologies and to facilitate the development of next-generation information technology products for the world market.

The National Research Council, Canada's leading science and technology agency, develops knowledge through its own basic and directed research programs. NRC also provides a wide range of services, facilities, technology transfer programs and collaborative research opportunities to help Canadian industry maintain its high standards of excellence and international competitiveness.

For more information contact:
Janice Nurski, Information Services (613) 993-4806

ministere des Communications (MDC). Il y a egalement eu des consultations etendues avec des representants de la clientele utilisatrice.

Les trois organismes federaux precedemment cites collaborent aussi a une etude, commanditee part ISTC, ayant pour objet de determiner la faisabilite de la mise en place d'un reseau national multimedia et a large bande pour developper la recherche cooperative dans le domaine des technologies strategiques et pour faciliter la mise au point, pour les marches mondiaux, des produits de la technologie informatique de la prochaine generation.

Principal organisme de recherche scientifique et technique du Canada, le Conseil national de recherches (CNRC) elargit le patrimone de nos connaissances grace a ses propres programmes de recherche fondamentale et thematique. Il offre egalement une vat gamme del services, d'installations, de programmmes de transfert de technologie et de recherche conjointe pour aider L'industrie canadienne a maintenir son haut niveau d'excellence et sa competitivite sur les marches internationaux.

Pour complement d'information:
Janice Nurski, Services de l'information (613) 993-4806

Supporting a World-Class Institution

by John Godfrey

Fraser Mustard wants $100 million, and he expects $50 million of it from you. He wants some companies to pledge $5 million each, and another 200 companies to come up with $5,000 apiece annually. He expects the federal government to come up with a matching $50 million. He deserves the money.

Mustard wants to establish an endowment fund to yield $7 million to support the work of the Canadian Institute for Advanced Research, an institution he helped create seven years ago. Mustard passionately believes that the institute is one of the few counterweights to the increasing fragmentation of Canada, and that it is tackling the large questions on which the future of Canada depends.

He argues that the key for economic growth in modern industrial states is science-based innovation. Only continuous innovation will increase the supply of tradeable goods and services on which the prosperity of Canada rests. One of the difficulties facing Canada is an excessive emphasis on short-term profits at the expense of the long-term investments necessary to build up the tradeable goods and services sector.

The institute is undertaking seven programs, and one of them, appropriately enough, is examining the role of innovation, which accounts for 80% of economic growth. Richard Lipsey, professor of economics at Simon Fraser University, heads up the project, which will draw on scholars from across Canada and from around the world. One of these is Brian Arthur from Stanford University in California, who notes that "conventional economic theory does not realistically mirror the economy driven by science-based innovation. Policies based on unrealistic theories will make it increasingly difficult to compete in the new global economy."

It is this kind of basic thinking about big and important questions that characterizes the work of the institute. Another program is artificial intelligence and robotics, which has just undergone a five-year review. Thanks to the institute, Canada has established three of the 20 top world-class centres in the area.

Created consortium

The institue has gone beyond studying the problem. It helped create PRECARN Associates Inc., a research consortium led by industry to undertake projects in long-term applied research in intelligent systems and advanced robotics. There are 33 companies involved, both producers and consumers.

The institute's work in this field is developing a transformational technology that can have applications for many diffent industries in Canada.

The Institute chooses projects that are on the frontiers of science, and where advances will have a catalytic intellectural or economic consequence. One of these is the program in cosmology, the study of the universe. As the description of the program notes, "the field of cosmology thus covers an immense amount of intellectual territory, from the forefront of astronomical observations to the deepest questions of philosophy."

Thanks to the work of the institue, Canada is a world centre in cosmology. The program has attracted outstanding young cosmologists to work at the Canadian Institute for Theoretical Astrophysics in Toronto. In Western Canada, the program draws on the work of two leading scientists in the area of gravity.

The institute's other scientific programs are in the critical areas of evolutionary biology and superconductivity. But its work also extennds into the social sciences, such as the previously mentioned program in economic growth, and two other projects in the health of populations and law and society.

Why should Canadian corporations and foundations get behind the work of the institute? Because the research being undertaken will have major consequences, directly and indirectly, for Canadian business. Consider the number of manufacturing firms that would profit from exposure to frontier research in artificial intelligence and robotics. Thanks to the institute, Canadian companies can have a privileged view of the latest developments in superconductivity.

Insurance companies, health-care institutions and, indeed, any corporation concerned about the effects of stress and the environment on the health of their employees should be closely monitoring (and suppporting) the health project. All of us should be paying attention to the economics and law and society projects.

In short, if Canada wishes to escape the economic, scientific and social constraints implicit in our current way of doing things, the best investment any of us could make in our future would be to help Fraser Mustard get his $100 million.

Reprinted with permission – Financial Post, November 6, 1989.

Mellon Foundation Supports International Text Encoding Project With $100,000 Grant

News Release

The Association for Computers and the Humanities, the Association for Computational Linguistics, and the Association for Literary and Linguistic Computing are pleased to announce that The Andrew W. Mellon Foundation has awarded a two-year $100,000 grant to support the Text Encoding Initiative (TEI). The TEI, which is jointly sponsored by these three organizations, is a major international project to develop guidelines for the preparation and exchange of machine-readable texts for scholarly research and to satisfy a broad range of uses by the language industries.

The project is being undertaken in response to the pressing need for a common text encoding scheme, demonstrated by the present chaotic diversity of formats now in use. The availability of these guidelines will make it possible for research groups to share data collections, which are both costly and time-consuming to develop.

Over 50 scholars from North America, Europe, and the Middle East are involved in TEI's effort to create sets of tags for marking features of texts. The tagsets, coded in the framework provided by the Standard Generalized Markup Language (SGML), will provide the means to mark physical features of text such as character sets and page layout. They will also provide discipline-specific tag sets to mark the results of research on the text, such as the analysis of sentence syntax or the identification of the metrical structure of verse.

Representatives of 15 scholarly and professional organizations form an Advisory Board for the TEI, in order to ensure that all of the needs and interests of the research community are adequately addressed.

The planning phase of this project was inaugurated by a $20,000 grant from the United States National Endowment for the Humanities, which later awarded a $185,000 grant to implement the first two years of a four-year work plan to produce the encoding guidelines. The TEI has also received a $100,000 grant from the European Economic Community.

La fondation Mellon appuit le projet d'encodage de texte international au moyen d'une subvention de 100 000 $.

L'Association pour les ordinateurs et les humanités, l'Association pour la linguistique informatique et l'Association pour la littérature et la linguistique informatique ont le plaisir d'annoncer que la fondation Andrew W. Mellon a accordé une subvention de 100 000 $ d'une durée de deux ans à l'appui de l'initiative d'encodage de texte (TEI, d'après l'acronyme anglais). TEI, qui est parainné conjointement par ces trois organisations, est un projet international d'envergure ayant pour but la mise au point de directives pour la préparation et l'échange de textes informatisés dans le cadre de la recherche scientifique tout en satisfaisant un large éventail d'utilisations par l'industrie de la langue.

Le projet a été entrepris en réponse au besoin pressant d'un shéma commun d'encodage de texte, mis en évidence par la diversité chaotique des formats présentement en cours. La présence de ces directives permettra aux groupes de recherche de partager des collections de données, lesquelles sont onéreuses à développer en dollars et en temps.

Au-delà de cinquante scientifiques d'Amérique du nord, d'Europe et du Proche-Orient participent aux efforts de TEI à la création densembles d'étiquettes pour marquer les caractéristiques des textes. Ces ensembles d'étiquettes, encodés dans le cadre du Standard Generalized Markup Language (SGML), faciliteront le marquage des caractéristiques physiques des textes, telles les jeux de caractères et la mise en page. Des ensembles d'étiquettes pour des disciplines donnés permettront également de marquer le résultat de recherches sur les textes, telles l'analyse syntaxique des phrases ou l'indentification de la structure métrique des vers.

Un conseil consulatif pour ETI formé de représentants de quinze organisations scientifiques et professionnelles veille à assurer que tous les besoins et intérêts de la communité de rechereche soient servis adéquatement.

Le période de plantification de ce projet a été démarré grâce à une subvention de 20 000 $ de la part de la Dotation nationale américaine pour les humanités, laquelle a par la suite accordé une subvention de 185 000 $ pour les deux premirès années d'un plan de quatre ans visant à produire les directives d'encodage. TEI a aussi reçu une subvention de 100 000 $ de la Communauté économique européenne.

A Theory of Mind

Aristotle in his **De Anima** described mind in Greek as η πρωτη εντελεχεια σωματοσ δυναμει ζωην εχοντοσ - "the first entelechy of a body potentially having life."

Mentifex Systems has studied the brain and the mind in the manner and tradition of an ancient philosopher - eclectically and with a driving thirst for knowledge and enlightenment. The result is a uniquely original theory of mind (theory - from the Greek θεωρια, a way of looking at things).

At **Metifex** we have used the tools of logic and linguistics to look at the mind as a black box, an *ignotum quid*, something unknown. We looked for fourteen years before we would cry ευρηκα - eureka.

What we found in our heuristics was a way of tying it all together, of banishing the *homunculus* and of tracing out the flows of information in a mind.

Meanwhile, the technology of neural nets has come up alongside us. We believe that we have something to offer you who have advanced the state of the art in neural nets. Our **Mentifex** design is an *a priori* neural network design, based as it is upon the preliminary groundwork of Eccles and of Hubel and Wiesel, and upon the linguistics of Chomsky. But we have left the *terra firma* of proven research and of demonstrable results in our headlong rush to arrive at a comprehensive theory of mind, a working paradigm for the construction of an artificial intelligence.

An unexamined mind-model is not worth having. Pease write and ask for the details of our design, and then examine it. Our resources are too limited to develop the design, while you perhaps have the resources but no algorithm.

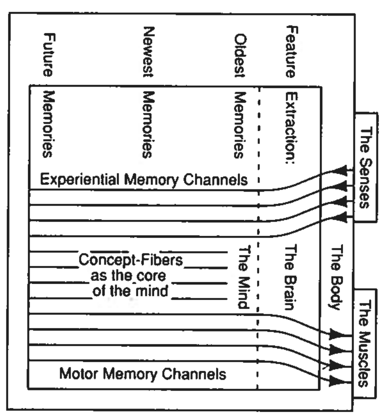
We especially wish to share our thoughts with fellow eccentrics, with non-conforming free-thinking individuals engaged in speculations similar to our own. If you are open to new ideas and curious about φιλοσοφια βιου κυβερνητησ, please make contact.

The superficial figure depicted above only hints at the full depth of our mind-model. Before we could cry "Ευρηκα!", our system-diagram was much more complicated - and wrong. The closer we approached to our present theory of mind, the simpler and more compelling our synthesis became. Probably it is still wrong, but we would like to share it with you nonetheless. Pick it full of holes and free us of our delusion of being *mentifices* - mindmakers.

If you don't like to write away for things and you would prefer to judge the **Mentifex** mind-model solely on the basis of information already at hand, here are the crucial details from which you can perhaps reconstruct the design in your own image.

An advancing front of consciousness moves slowly down the lifelong memory channels depicted here. Each fleeting moment of the present has its own vast neural net woven across the mindgrid. The abstract central core of the mind imparts a time-dimension to the massively parallel network. Within that abstract core, hierarchical grammar structures arise to control a vocabulary of words, or symbols, deposited over time as engrams in the auditory memory channel. Whether the engrams are stored as holograms or as synaptic conjunctions or as interference patterns, the Chomskyan grammar rules still hold sway over the associative meanderings of conceptual information in the mind.

MENTIFEX SYSTEM
Post Office Box 31326
Seattle, WA 98103-1326



Vision is accounted for through the feature extraction of Hubel and Wiesel, but, since vision is superfluous to an intelligent human mind, you are not permitted to debunk the **Mentifex** paradigm on the basis of a faulty subsystem for vision.

Motor control is discussed extensively in the documents which are yours for the asking, but, since locomotion is more of the body than of the mind, a repudiation of the provisions for motor behavior will not collapse the mentifical edifice.

Language is the central theater of this theory of mind. If you can debunk or disprove our observations about language, then we doff our Comskyan hats to you and we thank you for setting us back upon our philosophic quest in search of an answer to the ancient imperative: ? - "Know Thyself!"

User Modeling and User-adapted Interaction: An International Journal

A new journal has been founded that provides an interdisciplinary forum for the dissemination of new research results on all aspects of user-adapted interaction in person-machine interfaces, natural-language dialog systems, intelligent tutoring systems and intelligent interfaces. The journal publishes high-quality original papers contributing to these fields, including the following areas:

- acquisition of user and student models
- conceptual models, mental models
- dialog planning and response tailoring
- levels of user expertise
- explanation strategies
- plan recognition and generation
- presentation planning
- recognition and correction of misconceptions
- user stereotypes
- formal representation of user and student models
- shell systems for user modeling
- tutoring strategies

Relevant papers from the fields of Psychology, Linguistics and the Instructional Sciences are also considered.

The central audience of the journal are researchers, students and industrial practitioners from the following areas: Artificial Intelligence (focus on knowledge-based systems), Human-Computer Interaction (focus on cognitive engineering and intelligent interfaces), Linguistics (focuses on pragmatics and dialog models), and the Instructional Sciences (focus on computer-based tutoring systems).

Editor:

Alfred Kobsa
Dept. of Computer Science
University of Saarbruecken
D-6600 Saarbruecken 11
WEST GERMANY

Editorial Board:

UMUAI is published quarterly by Kluwer Academic Publishers, Dordrecht, Netherlands. The first issue is to appear in January, 1991.

Authors who wish to submit a paper should send four copies to the Editor. Electronic submission of LaTeX files is also possible if the special UMUAI style is used. For further information (especially on electronic submission), send a message to umai-info%fb10vax.informatik.uni-saarland.ddbp.de@relay.cs.net or ((uunet, mcvax)!)unido!sbsvax!fb10vax!umai-info. These addresses will be probably changed to umuai-info@cs.uni-sb.de in 1990.

"Simulation: Reality in the 90s"

1990 Summer Computer Simulation Conference - Calgary, Canada

The 1990 Summer Computer Simulation Conference features innovative presentations, panel discussions, state-of-the-art reviews, tutorials, and trade show exhibits designed to provide a comprehensive coverage of the field of computer simulation.

Calgary Convention Center, Calgary, Canada

July 16 - 18, 1990 (immediately following the Calgary Exhibition & Stampede)

Conference Program

Physical, Chemical, & Engineering Applications
Marilyn B. Kloss, Mitchell and Gauthier, Associates

Computer Systems
Ragnar Nilson, Simulation Services

Intelligent Simulation Environments
Joe K. Clema, IIT Research Institute

Computer Integrated Manufacturing
Douglas H. Norrie, University Of Calgary

Energy & Environmental Sciences
Antonin Settari, Simtech Consulting Services

Advanced Information Processing
Eugene J.H. Kerckhoffs, Delft University

Transportation Systems
Roger Varadarajan, GM Systems Engineering Center

Government, Management, & Social Sciences
Charles I. Bartfeld, The American University
Celik Parkan, University of Calgary

Simulation Methodologies
Ralph C. Huntsinger, Cal. State University, Chico

Artificial Intelligence
Brian Gaines, University of Calgary

Training Simulators
Alain Stevens, Combustion Engineering

CAD: Process & Control Systems
Reg Wood, University of Alberta

Radar & Communications
Buckley Pierstorff, Mitre Corporation
William D. Wade, Wade Engineering

Biomedical Sciences
Ken Strom, GM Research Laboratories

Strategic Defense Initiative
William Waite, AEgis Research Corp.

Aerospace & Missiles
Robin Kirkham, Lockheed/ESC
Mitchell E. Sisle, Raytheon

Agricultural
Basil Acock, USDA Agricultural Research Services

Professional Development Tutorials

SCSC'90 offers an extensive professional development program of introductory and advanced tutorials. Each tutorial runs for 1 1/2 hours. Tutorial fees are C$75 each. Where there are two sessions in the tutorial, the fee for attending both is C$120. Please note that *advance registration* is required for the professional development program. To register, contact the registration chair at the address below. The tutorials include:

"Artificial Intelligence (Introduction and Knowledge Acquisition)"
Mildred Shaw and Bruce MacDonald, University of Calgary

"Configuration Management (H/W and S/W)"
L.G. Egan, Software Certification Institute

"Recent Developments in Security for Distributed Systems"
Harold Podell, Marshall Abrams and Grace Hammonds, Computer Educators Inc.

"Neural Networks"
Mary Lou Padgett, Auburn University

"Performance Analysis and Logic Verfication using PetriNets"
John Brodie and Janice Snow, Simulogics, Inc.

"Discrete Event Simulation of Concurrency"
Dennis Mulcare, Lockheed Aeronautical Systems

"New Developments in Continuous System Simulation Languages"
Ralph Huntsinger, California State University, Chico

Trade Show & Exhibits

Leading commercial vendors will participate in the conference trade show & exhibition. Early exhibitors include:
Apollo Computer, Control Data, Sun Microsystems, Jade Simulations, Meiko Scientific, Pulsearch Consolidated Technology, Pulsonic Geophysical Ltd.

1990 SCSC Committee

Business Chair
Greg Kletke, Jade Simulations

Exhibits Chair
Janet McCracken, Alberta Research Council
Ena Spalding, Alberta Research Council

Publicity Chair
Carmen Colborne, Jade Simulations
Debbie Leishman, National Research Council

General Chair
Dr. Brian Unger, Jade Simulations

Program Chair
William Y. Svrcek, University of Calgary

Assistant Program Chair
John R. McRae, University of Calgary

Associate Program Chair
Edward C. Roche, New Jersey Institute of Technology

Registration Chair
Kevin F. O'Connor, Dataspan Technology Inc.

Companions Program
Dianne Van Dyl, Jade Simulations

Special Programs Chair
Tandy McGibbon, Government of Alberta

Local Arrangements
Michele Goshulak, Government of Canada

Sponsored by...
The Society for Computer Simulation
P.O. Box 17900, San Diego, CA 92117
Phone: (619) 277-3888 Fax: (619) 277-3930

For registration information, contact
Camille Sinanan
Computer Science Dept., University of Calgary
Phone: (403) 220-6316 Fax: (403) 284-4707

Marlene Jones: Prescription for a healthier Canadian AI community

by Connie Bryson

After two years as senior editor of Canadian Artificial Intelligence magazine, Marlene Jones is moving on to new challenges in both her work and personal life. Professionally, she is one of the organizers of a NATO research institute, "Syntheses of Instructional Science and Computing Science for effective instructional computing systems", to be held in mid July in Calgary. On the home front, she and her spouse Dan are expecting their first child in February.

It took some convincing for Jones to agree to the interview but after arm-twisting from Roy Masrani, co-editor of the magazine, she finally gave in. Jones felt it was too soon to have a feature article on a newly "retired" editor. But there's an advantage to immediacy. As you'll read in the interview, Jones' comments and suggestions about the magazine, CSCSI, CIPS, and the Canadian AI community are lively and invigorating.

Of course there's more to Marlene Jones' career than Canadian AI magazine and the first part of this interview focusses on her "real" work - - AI and Education. She has a PhD (Computer Science) from the University of Toronto and an MEd (Education of Exceptional Children) from the University of Saskatchewan. Prior to joining the Alberta Research Council as a Senior Researcher in Artificial Intelligence in 1987, she was a tenured Associate Professor at the University of Waterloo.

Canadian AI: Where and when did your interest in AI and education begin?

Jones: While I was doing my PhD at U of T, I started to pursue another interest — children with learning disabilities. I went to OISE (Ontario Institute for Studies in Education) and took a graduate course in the area and found I enjoyed it immensely. I also found that OISE was a good place to retreat from working on my thesis. I'd go up there and spend the afternoon reading in the library.

Education turned out to be more than a passing interest. I got more and more involved in the area and decided to pursue it after my PhD. I had planned to take an academic position right away and do a Master's degree part-time in special education. But one of the pieces of advice I got from academics, particularly new professors, was that in their first year they spent almost all their time preparing lectures. The more I thought about it, the more I realized I should go to school full-time. And so I enrolled in the Master's program in education at the University of Saskatchewan.

When I was finishing up my PhD, I was hesitant to tell people I was going to do a Master's in education because education has always been viewed as a very soft area. A lot of people criticize education research as unsound, not adhering to good experimental techniques. But that's not true. I'm not denying that there is some poor research done in Education but that's true of any area.

After a year of course work, I started working in AI and education. It was my interest in education that led me into AI and education, rather than going in from computer science. I found it was a natural transition.

Canadian AI: Surely it was more like a drastic switch?

Jones: Well it made a lot of sense to me, but I guess that viewed from the outside it was quite a change in academic area.

I think because I was so enthusiastic, I didn't have much difficulty switching areas. One has to pick up a substantial amount of background but I gave myself a year to take courses before I launched into a project.

I haven't looked at the work I did on combinatorial algorithms for over four years now. For a while I kept working in that area as well as AI and Education, mainly becuase I enjoyed the people I worked with. It took a few years to convince myself that I had to give up one of the areas.

Even though I do not publish in the area of combinatorial algorithms anymore, I feel my theoretical background has stood me in good stead.

Canadian AI: You've said that researchers in AI and Education have been good at making promises but not as good at delivering workable systems. Is delivery beginning to catch up with the promises?

Jones: There's been a major improvement in the last few years; several things have changed.

In an editorial in Computational Intelligence in 1986, I said we needed closer collaboration between the AI researchers in the area of AI and Education and the educators — education psychologists, instructional designers, and so on. What we've seen in the last couple of years are conferences that attract both groups. That's one critical step.

The NATO research institute on Instructional Computing Systems this summer (July 15-27 in Calgary) is an example of the increased collaboration that's been happening. Dr. Phil Winne of Simon Fraser University and I are the two directors of the institute. The two other organizers are Elliot Soloway (U of Michigan) and Peter Goodyear (U of Lancaster).

The purpose of the workshop is to try to bridge the gap between educators and ITS (Intelligent Tutoring Systems) researchers. It's an opportunity for both groups to talk about what is needed, discuss research problems, and advance solutions.

For me, this workshop is what I've been waiting for. I've always made an effort to cross the boundary because I know quite a few educators and education researchers and feel comfortable talking to them. But there's so much to know in the education area that I'd have to take more graduate courses to really get up to speed. I'm not going to have that luxury again.

So the best thing we can do is get together a group of people — about 70 — with common interests and different backgrounds. I think we'll cover a lot of ground in that time, it should be really exciting.

Connie Bryson is a free-lance technical writer based in Vegreville, Alberta.

Another change in the area of AI and Education is that there are now researchers who have both computer science and education backgrounds. When I first got into this area, I really felt like the odd person out. But now when I go to conferences, I don't feel that way.

Canadian AI: Aside from holding joint conferences, is there much interaction among Canadian AI and education researchers?

Jones: When you look across Canada, ITS researchers are fairly spread out. There are small pockets — Montreal, Toronto, Saskatoon, Calgary, Vancouver — and no one group is very large. That's a handicap.

Networking is up to us but there are things that could be done to help. For example, I'm an adjunct professor at the University of Saskatchewan, an obvious place for me to be because of similar interests and the close ties I have with some of the researchers. The problem I've encountered is finding appropriate funding so I can get back and forth from Saskatchewan. My NSERC grant forbids me to use those funds for travel to Saskatoon.

It bothers me that restricting the use of my grant restricts the collaboration that can be done. The interaction isn't frivilous. We can talk over the phone or talk by e-mail but it's not the same thing as spending several hours or days across a table.

Canadian AI: Have the changes in ITS research had an effect on the commercial side of ITS?

Jones: There are now ITS systems in everyday use, not just in laboratories. There are success stories. However, it's still relatively rare to see a system developed by a mainstream AI and Education researcher migrate out of the lab. Two researchers who have done it very successfullly are Bev Woolf and John Anderson.

There's a fundamental gap between what is going on in that lab and what makes it into the classroom. There's an excellent report on this problem, produced in 1988 by Elliot Soloway and Roy Pea for the US Office of Technology Assessment. The authors identified three major gaps — the knowledge transfer gap, the research practice gap, and the technology transfer gap.

They said the underlying problem that causes these gaps is our paradigm of the transmission of research into practice. How the results come down from the researchers — whether they are education researchers or ITS researchers — to teacher educators, and eventually into practice. Or how the research results trickle down from the researchers to instructional software developers, or to training groups in corporations, and on to the end user.

This "trickling down" process is far too slow. We need to bridge the gap between what's going on in the lab and what's happening in the classroom. There's also the gap between what's going on in the research lab and what software companies are developing. In their report, Pea and Soloway advocate the creation of Centers for Interactive Technologies in Education.

But there's a lot we can do without waiting for the government to fund major research centres. One of the things is closer collaboration between researchers and the end user. That's one of the reasons I'm working at the Alberta Research Council.

But in general, I think the research community doesn't yet see itself as part of the problem. There's some interest on the part of ITS researchers to start working with companies but I often get the feeling it's just another way of getting research funding rather than a genuine change of heart. And until technology transfer is one of the fundamental things researchers want to accomplish, we're still going to have relatively little impact on how technology is used or gets incorporated. There's a lot of excellent research going on but we need to make an effort to prove to the end users that this research is really practical.

When I go to talk to a company about their training needs, they don't want to hear that I can build them a wonderful ITS system but it's going to take me two years to do it. They don't want to hear about the wonders of student modelling and in the next breadth hear that the performance might be slow because of the amount of inferencing that is required within the system. They want to know what I can produce as a prototype in a few months. And I think we have to do that more often.

I'm not against curiosity-based research, that kind of research is very valid. I'm saying that we also need to work with corporations, software companies and educational institutions.

Canadian AI: Was that one of the reasons you decided to leave that that was becoming the goal of doing research, rather than the end product or the exploration?

Jones: I also found I wasn't getting enough sense of the concrete. It wasn't sufficient to know that my ideas got published, or even that someone had built on those ideas and that resulted in publication. I wanted to see the ideas in action, I wanted to build systems that made it out into the classroom or corporation. That was the main reason I went to the Alberta Research Council (ARC). It provided precisely that environment.

Working here gives me the chance to keep ties with my academic colleagues, stay up on the research, but also work with companies and actually see systems develop and ultimately get used. It doesn't matter to me if all the development isn't done under my direction, I just want to know that whoever is working on it will see that system through to completion.

It's important to me to work with the end users and this is something I've been able to do at ARC. Typical ITS experimentation in an academic setting involves using your second- or third-year students. I wanted to be working in situations where I had more exposure to the "real world" and a better chance of finding out what the teachers and students want and don't want.

Working with companies that are surviving in AI and Education really drums home the practical problems, particularly problems with performance and storage issues. You don't have to face a lot of these things in the research environment. I've enjoyed having to face them.

Canadian AI: Why did you become involved with Canadian AI magazine?

Jones: I agreed to edit the magazine because I believe that everyone has to take a turn and be active in CSCSI. I had been approached previously but I was doing a lot of travelling at the time and declined. When I was asked a year later to take on the magazine I felt that it was my turn to do something.

I also felt that being at ARC was an advantage because ARC was willing to lend support. The people in the Advanced Technologies Department are very enthusiastic. At times the volunteers were far more gung-ho than I could manage to be.

The first year was a lot of fun — a lot of hard work, but fun. With each issue I wondered how Graeme (Hirst) had managed to do this on his own. I don't think he's got the recognition he deserves for doing the magazine as long as he did and doing such a professional job.

But every issue brought more frustrations. For one, you can't count on material to arrive on time or to arrive at all. And it's still a relatively small portion of the membership that gets actively involved in sending in submissions.

There's also a lack of involvement by graduate students across Canada. I was delighted to see in the last issue that a senior graduate student wrote the U of W research report. I think we better start tapping the graduate students even more; they're often the ones who have more drive. That means we have to start encouraging graduate students to get involved in CSCSI.

Canadian AI: Why is it important to have a Canadian AI magazine?

Jones: Without the magazine, I wouldn't know as much about why my Canadian colleagues are doing. I would know what my Canadian colleagues in AI and Education are doing, but I wouldn't be as well-informed about current research at the various universities.

I think it's very important to be knowledgable about what the Canadian AI community is doing. If you know what others are doing, you're more apt to talk about what they're doing. And I think we all need to do more of that.

Canadian AI magazine also carries articles on more global issues such as AI in the resource industries, Precarn, CIAR, Canada's space industry. Those are the kinds of things I want to know about and I'm not going to find out about them reading American publications. Canadian AI magazine is where these things get pulled together.

Canadian AI: Is the future of the magazine tied to the future of CSCSI?

Jones: Yes, and that's another one of the frustrations. CSCSI has a relatively low profile. It could be a strong lobby group, but isn't. I don't know whether that's a reflection of the Canadian personality or simply due to the fact that we're all very, very busy and things like CSCSI never become a high priority. They are the things that can be put off until tomorrow.

I think that if the society continues to be totally volunteer-based, it's not going to flourish. I feel the executive has to take drastic steps. A year and a half ago I suggested the society consider corporate sponsorship. That was basically put on the back burner and now it's being raised again.

There are corporations out there that would be interested in sponsoring the society. We need to go out and solicit these sponsorships and hire even a half-time paid employee to work on this.

And then there are the problems with our association with CIPS. Here's just one example. For every issue of the magazine, we have to get in touch with the membership contact at CIPS for an up-to-date mailing list. For every issue it's been a different person. There isn't any continuity.

Everytime I've raised concerns about these problems — people not getting their memberships renewed, losing memberships — the answer is always that the problems are being resolved. There are only so many times you can hear that answer before you don't listen.

I know members have found it very frustrating when all of a sudden they don't get the magazine anymore and yet they've paid their dues. Or they never received a renewal notice. And I can't blame them for getting so frustrated that they just stop belonging. CSCSI is a small organization, but I think we're smaller than we need be. If we could resolve some of our membership tracking problems, we would have more members.

My personal opinion is that CSCSI must break away from CIPS. But in order to do that we need a stronger financial basis and that comes back to what I said before. I think we need corporate sponsors and we've got to be willing to have a paid employee to look after memberships, advertising, sponsorships, etc. It's a gamble but it's one that I think the society must take if it is to flourish.

Meta-Programming = Meta-Interpretation + Meta-Unification

Georges-Henri Moll

RESUME: La Méta-programmation est une technique puissante utilisée en programmation logique. Elle a été jusqu'ici surtout utilisée pour concevoir des prototypes utilisant des stratégies nonstandard d'exploration d'arbres de résolution. Grâce à cette technique, il est possible de prototyper très rapidement des moteurs d'inférence. Il est également, et surtout, possible de séparer le contrôle des programmes. Les évaluateurs partiels sont eux aussi des méta-interprètes, et ils ont été utilisés dans différents domaines, tels l'apprentissage automatique, les bases de données données déductives, etc.

Mais un des aspects du controle semble avoir été laissé de côté jusqu'ici à savoir le contrôle d'unification. Nous présentons un prototype réaliste utilisant cette technique. Nous montrons également comment la méta-unification peut être utilisée seule, ou très facilement combinée avec la méta-interprétation classique.

MOTS CLE: Programmation logique, Bases de Données déductives, Relations imbriquées, Logique avec ensembles, Résolution, Unification, Méta-Interprétation.

Université Laval, Faculté des sciences et de génie, Département d'informatique G1K 7P4 Ste-Foy, Québec Canada

tel: (418) 656-5189 e-mail: moll@lavalvm1.bitnet
fax: (418) 656-5902 telex: 051-31621

1-Introduction

Meta-programming is a well-known programming technique of logic programming. The key idea is to use the standard PROLOG interpreter to write logic program interpreters, so that rewriting a backtracking strategy and a unifier becomes useless.

A sample meta-interpreter is given by the now classical three clause program [SHAP-86]:

```
solve (true).
solve ((A,B)):-
  solve (A), solve (B)
solve (A):-
  clause (A,B), solve (B).
```

(We use the Edinburg Prolog syntax)

This program is a PROLOG standard strategy interpreter written in PROLOG. To obtain an interpreter with a specific strategy, this program has to be modified. So, many interpreters with various strategies can be written with this simple technique. As a matter of fact, meta-programming is used as a control technique, that is to define a strategy for exploring a resolution tree. However, up to now, no such technique has been proposed to modify not the strategy but the standard unification built-in algorithm of PROLOG. This aim of this paper is to propose a technique, called meta-unification, to pilot the unification algorithm, that is to do unification control. This may be very very useful when the standard unification of PROLOG has to be modified. We argue that generally, it is not necessary to change the source of the PROLOG interpreter, but, on the contrary, a meta-programming technique can be used (for prototyping by example). In section 2, we present the basic concept of control unification, and a technique to process it by meta-unification. Mainly, this techniqque allows to intercept classical unification without any interaction on standard resolution (resolution strategy, built-in handling by the standard interpreter, range of the cut and son on). In section 3 we illustrate this technique by three toy applications. In section 4, we present a real prototype using meta-unification. It is the COL interpreter. COL [ABITE-87] is a logic language including (logically defined) function evaluation and sets. It was designed as a data manipulation programming language for deductive databases with nested relations. At last, in section 5, we show how to use meta-unification together with meta-interpretation, in a very simple way.

2-How to control unification

Unification appears at a very low level: Each time a clause is fired in Prolog, its variables and are renamed and unification of its head with the current subgoal to be solved is done. The problem is to prevent standard unification (so to delay it), and replace it by another strategy. Here we are not interested in changing the resolution strategy, but only the unification algorithm. (Of course, meta-interpretation can be combined with unification control, this will be the matter of section 5). But let us have a first look at unification control alone. Suppose that a given program P has to be controlled:

The rules `Rk` of `P` are of form

```
Hk:-Q1k, . . . , Qnk.
```

The rules have to be transformed to be controlled by the unification controller. The predicate s1 (for solve literal) is inserted.

```
head (Hk, ________) :-
  sl (Q1K , . . . , sl (Qnk).
```

`head` will allow the unification control for calling rules with a specific head form, and s1 controls the transformation of literals in queues of rules. A simple syntactic translater can change standard rules into this form. Additional parameters can be chosen for head, allowing preselections on heads of rules. A good choice for those parameters is the name of the predicate asnd its arity. Built-in predicates can be let without the sl controller (exept if unification has to be controlled on them also). `sl` is defined as follows:

```
1 sl (_literal):-
2  call((
3 pre_unification (_literal,
     _good_parameters),
4 clause (head(_candidate_literal,
   _good_parameters),_clause_queue),
5 unify (_literal,_candidate_literal),
     _clause_queue
7    )).
```

The purpose of `pre_unification` is to make restrictions on the possible heads of rules that are candidate to be unifiable with the current literal. The output of this predicate is a set of good parameters to be used to choose a rule the head of which is a good candidate (line 4). `pre_unification` must be consistent with unify, which effectively processes the unification. Lines 3 to 6 are fired via the `call` predicate. This allows to let the standard resolution strategy take place. Built-in are fired normally, especially the cut predicate, the range of which remains unchanged thanks to this trick. Precisely, the cut ranges up to the head of rule.

Note that predicates `pre_unification` and `unify` can possibly backtrack. If line 3 is erased and line 4 is replaced by

```
clause (head(_literal, _, _),
  _clause_queue),
```

if, moreover, unify is defined by

```
unify (_x, _x).
```

then the unification is the Prolog standard one.

3-Toy applications

Immaterial Arity

A possible application of this programming technique is to make arity immaterial for unification.
That means that unifying

```
p (t1, t2, . . . tn)
```

with

```
p (11, 12, . . . lm)
```

(where , tl to Il to tn & Il to IM are terms, cand supposing that m is superior to n), can be done by unifying tuple

`(t1, . . . , tn)` with `(11, . . . , 1n)`,

not taking into account terns `1n+1` to `1m`.

To do so, we propose the following program, supposing that the rules have been put under form:

```
head (Hk_name, _arity):-
  sl(Q1k) , . . .  , sl (Qnk).
```

. . . the program is

```
sl (_literal):-
  call((
    functor (_literal, _name, _),
  clause(
  head (_candidate_literal, _name, _),
      _clause_queue
unify (_literal, _candidate_literal),
  _clause_queue
  )).
```

`functor` is a Prolog built-in predicate giving the name and the arity of a given term. Functor works bidirectionally, i.e. it generates a terms with new variables when it is given a name and arity. We now have to define unification itself:

```
unify (_a, _b):-
  _a = . . (_na / _pa),
  _b = . . (_na / _pb),
  list_unif (pa_pb).

list_unif ((_t/_q), (t / _qq)):-
  list_unif (q, _qq).
list_unif ( )_).
list_unif (_, ( )).
```


To find out how, see page 43

As a matter of fact, this definition allows to take into account only the first n parameters of each term.

Application to function evaluation

A possible application of control unification is to allow terms to express functions (returning values). Suppose that a Prolog predicate processes this evaluation:

```
evaluate (_term, _value)
```

An interpreter which takes this feature into account can be written so:

```
head (Hk, _name, _arity):-
    sl (Q1k) , . . . , sl (Qnk).

sl (_literal):-
  call ( (
  functor (_literal, _name, _arity),
  clause (head (_candidate_literal,
     _name, _arity),
     _clause_queue),
unify (_literal,
     _candidate_literal),
  _clause_queue
    ) ) .
```

`Unify/2` is then itself defined as follows:

```
unify (_a, _b):-
  _a = . . (_n / _lpa),
  _b = . .  (_n / _lpb),
  list_unif (_lpa, _lpb)

list_unif ((X / Xs), (Y / Ys)):-
  eval (X, V),
  eval (Y, V),
  list _unif (Xs, Ys).

list _unif ( ( ), ( ) ).
```

This program supposes that functors are fully evaluated before unification. More complex unifiers can be written supposing that the evaluation of terms is not full.

Constraint propagation

Another possible application of this technique is constraint propagation management. Suppose that in a given program, all variables are replaced by a triplet specifying the variable name, and its attached set of input and output constraints.

```
(_variable, _input_constraints,
   _output_constraints)
```

Rules must be written taking into account this data structure.
A rule of form:
Original form

```
head(_x, . . . ):-
  q1(_x, . . .),
  q2 (_x, . . .),
  . . . ,
  qn(_x, . . . ).
```

. . . has to be rewritten under form:
Constraint form

```
head((_x, _co,_cn), . . . ):-
  q1 ((_x, _co, _c1), . . . ),
  q2 ((_x, _c1, _c2, . . . ),
  . . . ,
  qn ((_x, _c-1, _cn), . . . ).
```

Of course, this program itself will be transformed into the "unification control" one:
Constraint+unification control form

```
head ( (_x, _c0, _cn), . . .):-
  sl(     q1 ((_x, _c0, _c1), . . .) ),
  sl( q2 ((_x, _c1, _c2), . . . ) ),
  . . . ,
  sl(     qn ((_x, _cn-1, _cn), . . . ) ).
```

Note that both previous transformation can be processed automatically. Note also that this transformation affect terms recursively. Moreover, there must exist a constraint specifier, the role of which is to add a new constraint on a given variable.

Original form

```
constraint (_x, _new_constraint)
```

Constraint form

```
constraint ((_x, _c _in,_c_out),
        _new_constraint)
```

Constraint+unification control form

```
constraint ((_x, _c _in,_c_out),
  _new_constraint
```

(no `sl ( . . . )` added, because `constraint/2` can be now considered as a builit-in predicate. This predicate (`constraint/2`) has in charge to verify the consistency of constraints when a new one is added, and to fail in case of unconsistency. Now, when two variables have to be unified, say (`_x, _cxi, _cxo`) and (`_y, _cyi, _cyo`), this means not only to make the substitution {_x / _y}, but also the add constraints `_cxi` and `_cyi`. This work has to be done recursively when two terms are unified.

Here is a generic version of such an unifier:
Suppose that a given _literal has to be unified with a _head of rule

```
unify (_head, _literal):-
  _head= . . (_n / _p1),
  _literal = . . (_n / _p2),
  list _parameters_unif (_p1, _p2).

list_parameters_unif ((t / _q),
    (_tt / qq):-
  parameter_unif (_t, _tt),
  list _parameters_unif (_q, _qq).

list_parameters_unif (( ),( )).

parameter_unif ((_x, _cxi, _cxo),
    (_y, _cyi, _cxo)):-
  var_or_atomic (_x, _y),!,
  _x = _y,
  constraint_reduction (_cxi,
    _cyi, _cxo).
```

```
var_ or _atomic (_x, _y): -var (_x),!.
var_ or_atomic (_x, _y): -var (_y),!.
var_ or _atomic (_x, _y): -atomic (_x),!.
var _or_atomic (_x, _y):-atomic (_y),!.
```

constraint _reduction merges _x and _y's constraints into a new one. In next rules, compound terms are treated, leafs of terms being supposed to be triplets.

```
parameter_unif ((_x, _cxi, _cxo),
        (_y, _cyi, _cxo)):-
    functor (_x, _n, _a),
    functor (_y, _n, _a),
    _x = . . (_n / _px),
    _y = . . (_n / _py),
    list _parameters _unif (_px, _py).
```

Of course, `constraint_reduction/3` and `constraint/2` depend on the type of constraints that are imposed. This version of a constraint propagation manager is very inefficient. But one has to keep in mind that it is a meta-programming technique. It is useful for prototyping, but not for commercial applications. However, previous examples show the power of the technique and its adaptability. This technique has been applied for prototyping an interpreter of a logical language. This language, named COL [ABITE-87], includes evaluation function features. We present in next section.

4- A meta-unifier for real: The COL interpreter

4.0-Introduction

A paper of Serge Abiteboul and Stephane Grumbach /ABITE-87/ presents a Complex Object Logical Language, COL. This language is o-perationally defined in terms of forward-chaining fix-points under some stratification hypothesis. It has been effectively implemented by its authors for deductive database management systems including structuration features, name Non First Normal Form (NF2) relations. This implementation follows the theoretical ideads since it works with fix-points. But backward chaining is another avenue in the field of deductive databases. So came the idea to implement a COL backward chaining interpreter and its interface with databases. This section describes the prototype of the COL interpreter.

4.1-Preliminaries

For a formal presentation, the reader should refer to /ABITE-87/ The two basic constructors for building nested relations, namely tuple and set constructors, are present in COL. So called data-functions are introduced to logically define and manipulate (finite) sets.

The syntax

COL is a sub-language of a first order typed language. The vocabulary is made of:

- Constants and variables
- Connectors and quantifiers
- Equality and membership symbols
- Predicate symbols
- Tuple constructors (...,...,... ...)
- Set constructs (...,...,... ...)
- Data-functions f(...,...,... ...)

Data functions are assumed to be set-valued. Note that the original COL definition includes types. This feature has been abandonned for the backward chaining prototype. The terms are the usual first order logic terms, built with tuple, set and data-function constructors. Closed terms are terms with no variable nor data-function symbols. The positives literals are of form $R(t_1, \ldots, t_n)$, $t_1=t_2$ or $t_1 \in t_2$, where R is a predicate symbol and $t_1, \ldots, t_n$ are terms. Negative literals are $\neg L$ if L is a positive literal.

Now the Col language is defined from the previous first order syntax as Horn clause logic is defined from classical first order logic. The key point is the concept of atomic literal. An atomic literal is a literal of the form $(R(t_1, \ldots, t_n)$, or $t_1 \in F(t_2, \ldots, t_n)$ where R is a predicate symbol, F is a data-function symbol and $t_1, \ldots, t_n$ are terms.

A Col rule is an expression of the form

$$A \leftarrow L_1 \wedge \ldots \wedge L_n$$

where A is an atomic literal, and the L_i are literals.

Universe, Base, Interpretation

Once symbols are defined, the Universe U is said to be all possible closed terms. Given a program P (set of COL rules), the associated base Bp is the set of all possible closed atoms with the data-functions and the predicates of P, and the terms of U.

An interpretation is a finite subset of Bp.

Let θ be a ground substitution, I be an interpretation of P, x a variable.

$\theta_I x = \theta x$

θ_I is then recursively defined as usual on terms and literals.

The difference between inerpretations and vlauations is for data-functions:

$\theta_I F(t_1, \ldots, t_n) = \{a | [a \in F(\theta_I t_1, \ldots, \theta_I t_n) \in I)$

Satisfaction

In /ABITE-87/, the definition of satisfaction (denoted $\models$) and its negation (denoted $\not\models$) are defined as follows:

.For each closed positive literal,

$I \models P(b_1, \ldots, b_n)$ iff $P(b_1, \ldots, b_n) \in I$;

$I \models b_1=b_2$ iff $b_1=b_2$ is a tautology and

$I \models b_1 \in b_2$ iff $b_1 \in b_2$ is a tautology

.For each closed negative literal $\neg B$,

$I \models \neg B$ iff $I \not\models B$

.Let $r = A \leftarrow L_1 \wedge \ldots \wedge L_n$. Then $I \models r$ iff for each valuation θ_I such that for each i, $I \models \theta_I Li$, then $I \models \theta_I A$

.For each program P, $I \models P$ iff for each rule in P, $I \models r$.

A model of P is an interpretation satisfying P.

Stratification

Unfortunately, as for clauses with negations, a given COL logic program does not have a unique minimal model. To give a semantic to the COL language, the definition of a canonical minimal model is necessary. As for negation, this definition is done via stratification conditions. Then fix-points can be calculated at each strate. The last one gives the canonical minimal model. For the backward chaining prototype, stratification has been abandonned. As a matter of fact, it was design in a forward chaining spirit. Stratified programs have always a calculable (via fixpoints) interpretation. However, this constraint is sometimes too strong. Some unstratified programs also have a minimal causal model. But of course, loosing stratification means loosing good properties. Unstratified COL programs may loop in a strange way: they can loop on recursive calls of the unifier.

Expressive power of COL

The following examples are taken from [ABITE-87]

Of course COL can express nested relations since it includes the two basic set and tuple constructors. But the important point is restructuration.

Nesting

Let P be a 2 place flat predicate

$x \in F(y) \leftarrow P(x,y)$

$Q(F(y),y) \leftarrow$

Unnesting

Let Q be a strujctured predicate

$Q(1,\{a,b,c\})$

$Q(2,\{d,e,f\})$ for instance

$P(x,y) \leftarrow Q(x,A), y \in A$

defines P as the unnest of Q.

Note that the concept of functions is enough to define negation:

$t \in F(t) \leftarrow P(t)$

$A(t,F(t)) \leftarrow$

$Q(t) \leftarrow A(t,\{\})$

Q is equivalent to the negation of P. This shown the role of { } as a null value, and also the strong links between data-functions, negation, stratification, and sets.

4-2 The COL interpreter

This section describes our practicle implementation.

3-1 Introduction

The COL language provides a powerful and theoretically well defined formalism to treat structured objects in a first order logic framework. Even if it is defined in terms of forward chaining fix-points, backwarchaining have many attractive qualities: It allows logic programming features. For database applications, backward chaining, as a top down approach, is a very natural way to go from queries to answers/solutions. Of course, this implies to define an unification and a strategy for COL, compatible with its semantics, as it is defined above.

Backward-chaining seems to be the good approach for database querying. As a matter of fact, it allows to explore only useful rule for the answer. Instanciations are pushed down, reducing the exploration of the SLD tree. On the other hand, forward-chaining up to fix-point is a strategy that produces all possibles deduced facts from a given problem.

Structure of the meta-interpreter for COL, and its meta-unifier

The COL meta-interpreter is designed as a three part software. The fist part is the unifier: As COL includes functions valuated by sets, unificaiton must be able to evaluate functions within a COL term at unification time. So the COL unification is not standard Prolog unification. Moreover, within sets, the order is immaterial, and as a consequence this has to be taken into account: For example (1,2) unifies with (2,1). The second part is a function evaluator: COL functions have to be evaluated. Roughly, the evaluator will release the interpreter, calling the rules that define athe current function to be evaluated. Third part: the interpreter. The strategy is the same as for Prolog. i.e. depth first, left literal first. But the meta-interpreter has to take into account that the unificaiton is non standard. So the COL interpreter explicitly calls the COL unifier during the resolution. As explained above, the three parts are not independent:

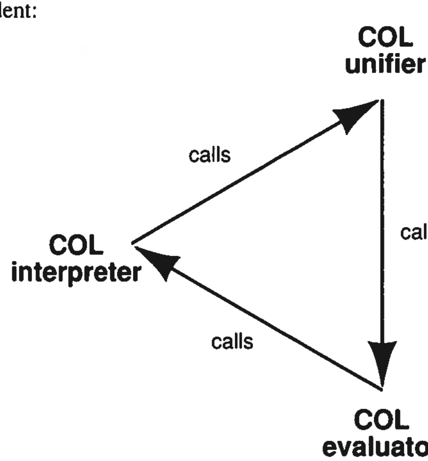


fig. 1: The three part software

The COL unifier

As described above, the COL language includes set-valuated functions. So the COL unifier must be able to deal with sets and functions. Sets are particular terms in the sense that the order of the elements is immaterial. Functions are particular terms in the sense that they have to be evaluated. The result of unification is as usual a set of variable substitution. So a two place predicate, namely "`unify (_x, _y)`" has been designed for COL. As standard Prolog unification, the tree (term) exploration is top-down, depth first and leftmost. Let's suppose that _x and _y are two terms to be unified:

Two special cases:

They can be predicative literals (of form P(t1, . . . , tn)), or so called function defining literals (of form t1 ∈ f(t2, . . . ,tn)). Those cases only occur during the resolution, when a current literal L of a given sub-goal must be unified with the head of a rule of the COL user's program. Note that a predicative literal p(t1, . . . ,tn) can only unify with a predicative literal of same name and arity p(t'1,. . . ,t'n). As usual, the result is obtained by successively unifying the ti with t'i, taking into account the propagation of the resulting substitution. Note that this propagation is done by the standard Prolog unification mechanism. The unification of functional literal t1 ∈ f(t2, . . . , tn) with t'1 ∈ f(t'2, . . . , t'n) is done the same way. Note however that the unification of t1 with t'1 is done at last, assuming that this term generally contains more functions to be evaluated than the t2 , . . . , tn. This must be considered only as an optimization hypothesis.

We now discuss of the unification of terms, name t and t', that is syntactical entities being possibly:

- constants
- variables
- tuples
- sets (empty or not)
- data functions

- The unification of a variable named t with anything (t') except a function is the substitution (t/t')
- Tuples only unify with tuples of same length:
  (t1 , . , tn) unifies with (t'1 , . , t'n) by successsively unifying the ti with t'i.
- Two sets A and B unify if:
  – They have same arity (this point will be discussed later)
  – The strategy for their unification is the following algorithm:

```
Choose an order for B=(b1 , . . . , bn)
For i=1 to n
Choose an element "a" from A
    (n possible choices)
  Unify a with b1
    the found m.g.u. is called q
  Unify (A/(a) )q with (b2 , . . . , bn)
End-for
```

- Unify a variable x with a function f:
  first evaluate f, the result is A; the m.g.u. is (x/A)
- To unify two functions f1 and f2; first evaluate them, the results are A1 and A2; unify A1 and A2
- To unify a function f with a nonempty set B; first evaluate f, the result is A; unify A with B
- To unify a function f with the empty set ø, verify that goal_x ∈ f has no solution (built-in predicate "\+")
- Note that the following cases of unification fail:
  – a function with a tuple or a constant
  – tuples, constant and sets do not unify with each others.

We have introduced in the COL interpreter a so called "lazy unification": In the case t and t' to be unified are strictly identical (built-in predicate "=="), nothing is done. This feature allows to sometimes avoid useless function evaluations, as well as sets permutations. Note that this "lazy unification" is safe in the sense that it doesn't prevent any needed function evaluation. No solutions are hidden, even in the case {_x, _y} to unify with {_x,_y} : unifier {_x/_y} is less general than mgu { }. Note also that the "==" test also includes constants unification.

SET UNIFICATION: a theoretical problem (not solved here)
Given two sets A and B, a "real-life" set unifier should return all variable substitutions θ such that Aθ=Bθ in the sense of set-equality. For reasons of combinatorial explosion, this has not been implemented so. We now present how it has been implemented. It has been said previously that a first condition for the unification of two sets was that they had same cardinality. This supposes that doubles are eliminated (or only counted once) from the sets, unless {1, 1,2} doesn't unify with {1,2}. Moreover, sets including variables are difficult to manage. For example, unifying {_x,_y} and {1} should succeed with substitution {_x/1, _y/1}. In the prototype, it fails. As sets are extensions, variables within sets are a bit paradoxal: extensions should include only ground terms. {_x,_y} doesn't express a particular set but all possible pairs and singletons, since variables in Horn clauses are universally quantified. As a consequence, its cardinality is not defined. So two problems arise for sets: doubles and variables.

The solutions proposed here are only practical (operationally defined) ones: The system never generates doubles and gives a message to the user when a "system-generated set" (generated by a data-function evaluation) is not ground. The user must be aware that the sets he produces must not include doubles (of course an automatic verification is always possible) and that variables in sets are dangerous. Note that types (in the sense of §II-1) are forgot: a variable can be unified with anything.

The COL interpreter

The SLD tree exploration strategy is the same as for standared Prolog. The only difference is unification. So the only job of the COL-interpreter is controlling unification, that is explicitly unify the current literal to be solved with a head of rule. Moreover, built-ins are authorized. In the prototype, we have limited the built-ins to:

```
-! (cut)
-write (_)
- _x is _y (arithmetic calculus)
```

but of course any built-in could be accepted (with very few modifications), even if the setof for example is useless for COL. In fact it has no sense to include setof/bagof in COL since the philosophy of this language is to define sets by rules.

On the contrary, the role of the cut is very important: any Prolog programmer is aware of it. In COL, the effect of the cut predicate is exactly the same as in Prolog. It is in fact implemented via the Prolog cut. Remember that the COL interpreter is a meta-interpreter. In COI the cut can be used to prevent a backtrack on a set unification, when for example the first one is enough. The "=" primitive is also available. It releases a COL unification. The meta-interpreter itself is very simple. It is mainly a call to the standard interpreter, however, unificaiton is controlled. That is why a "meta form" (facts) of the user's program is needed.

Let a COL rule be HEAD <- QUEUE. It is translated into

meta (NAME OF PREDICATE OR FUNCTION,
ARITY OF PREDICATE OR FUNCTION,
HEAD,
TRANSFORMED QUEQUE.

The queque is transformed into a form interpretable by Prolog. Let the QUEUE be the conjunction of literals I1, . . . , 1n. Ii may be a built-in, or a predicative or a functional literal. In both last cases, Ii is transformed into `sl (1i)`. Else it remains unchanged. This work can be called pre-formatting. Now let's define the sl predicate, which is the main predicate of the COL-metainterpreter.

```
1- sl (_x = _y):- unify (_x,_y).
2- sl(_t):-predicate (_t),!,
     call (case_predicate (_t, _q), ,_q).
3- sl (_t):- function (_t),!,
     call (case _function (_t, _q), _q).
4- sl (_x is in (_h / _q):-
     unify (_x, _h).
5- sl (_x is in (_h / _q)):-
     sl (_x is in _q).
```

"`isin`" stands for "∈"

Line 1 defines the case of "=" COL built-in, which must not be interpreted directly by Prolog.

Line 2 is the predicative literal case. "`case_predicate (_p, _q)`" COL-unifies `_p` with a head of rule of the user's program (in meta-form) and produces a queue `_q` to be solved.

Line 3 is the functional literal case, built on the same principle as line 2.

Lines 4 and 5 define the "`isin`" COL built-in.

The structure "`call (case_predicate (_t,_q),_q)`" allows to have the appropriate range for one (or more) possible "!" in `_q`.

The COL function evaluator:

The kernel is simply defined as follows:

```
evaluate (_f,_e):-
  setof (_x, sl (_x is in _f), _e.
```

In fact, the evaluator is a bit more complex, for the evaluation must:

– treat undefined function and evaluate them to Ø.

– treat defined function for which the goal _x ? _f has no solution. The result, in this case, is also the empty set.

– a warning is sent to the user a set with variable(s) is produced.

Note that the evaluator backtracks:

example: Let a COL program be

```
p (1,a).
p (2,b).
p (3,a).
_x is in f(_y):-p (_x,_y).
```

The evaluation of `f (_t)`, namely the goal:

```
?- _z = f(_t).
```

returns

```
_t = a
_z = (1,2)

_t = b
_z = (3)
```

Note that the goal:

```
?- _z = f (c)
```

returns

```
_z = nil
```

Another example: a function returning a set with a variable:

Let a COL program be

```
  _x is in f0A.
/*f0A is the full universe */
```

The evaluation of f0A returns `{_x}`, which is not very logical but returning the full universe of possible terms is impossible). That is why a message is sent to the user.

The expressive power of "backward-COL"

It is the same as Prolog's one (including a set of/bagof and built-ins). It is the fact the same as a Von Neumann machine, like every sequential programming language. But it is a language of higher level than Prolog in the sense that the "`setof`" is masked to the user by the facility to define sets via rules. As a consequence, the current variable defining a set is at the same level (universally quantified) as the other ones, which is not the case for "`setof`". In

```
. . . , set of(_x,p(_x,_y),_1) , . . .
```

the `_x` variable is not universally quantified, and so cannot be used in other literals of the rule without any strange effects. In fact, it is local to the setof predicate. So a bad programmer could affect _x before calling set of `(_x,p(_x, _y),_1)`, by a expression of form `foobar (_x) , . . . . . setof (_x,p`

(_x, _y), _1). Of course, this has no meaning since variable _x should be local to the setof expression.
On the contrary, in the COL rule

```
_x isin a (_y):- p(_x,_y).
```

the _x variable is universally quantified, the produced set is name a and reusable in any rule of the COL user's program. In this case, variable _x is local to the rule but not only to a part of a Prolog rule. This has been obtained by separating the part of the rule which processes the building of the set in a special rule. As a consequence, the user has been forced to give a name to the function that represents this grouping (namely a (_)). Now, using a (_), the user has no more references to the nesting variable _x. Once a (_) available, the nested version of P can be defined as follows:

```
G (a(_y),_y).
```

Moreover, in COL, the system manages sets, i.e. the user has not to deal explicitly with the possible permutations of variables.

Theoretical aspects

We have not proved that the strategy of the interpreter is compatible with the canonical semantics defined in [ABITE-87]. That is, given a program P and a query Q, does the system give solutions of the minimal canonical causal model (this is the validation problem) and does it give all the solutions of this model (this is the completeness problem). It may happen that the underlying model of the defined backward strategy is not the same as the one of forward chaining, for there is not only one minimal causal model. But the canonical one seems to be the most natural one. Those theoretical aspects are surely important. However an operationally defined semantics is temporarily acceptable.

Function evaluation

Up to now, functions are evaluated "when needed". This looks like a lazy evaluation strategy. "When needed" means in fact that the evaluation of a data-function is released when it has to be unified. However, a resolution can give as results not fully evaluated terms.
For example, goal

```
?- _x = (f(1), f(2))
```

returns substitution {_x / (f(1), f (2))}

On the contrary

```
?- _x = f (1)
```

produces an evaluation of f(1). This is an arbitrary choice, in fact to give the user the possibility of evaluating explicitly a function via unification. Of course the "when needed" strategy of function evaluation is available during the resolution. For example, in the first example (and in fact even in the second one) the evaluation of f(1) and f(2) are not needed. The user, however, may find strange to find unevaluated functions in the answer of the COL-interpreter. A possible solution is to end each resolution (been calculated by a "when needed strategy" where for example _x unifies with f (1) with substitution {_x/f(1)}) by a full evaluation of the terms given as answers to the users (if in previous case the user asked for _x, then f (1) is evaluated at last step). Theoretical aspects of function evaluation have not been taken into account up to now. A "full" evaluation at the end of each resolution would mean some mix of bottom-up and top-down strategies. Moreover, optimizations could be taken into such as avoiding to evaluate the same functional expression twice.

Coupling COL with RDBMS

As a matter of fact, COL [ABITE-87] has been designed to be a logic programming langugage for databases, being able to manage complex (structured) data. A COL forward chaining interpreter has been effectively implemented as a data-manipulation language at INRIA (France). The COL backward chaining meta-interpreter has also been coupled with a relational database:
The main choices for the COL/RDMS coupling are:

- transparency:
  the user sees the database tuples as if they were Prolog facts.
- COL with builtins
- loose coupling: coupling via the standard data-manipulation language (namely SQL). This feature allows a very flexible use of the interface: the COL shell can be added to any existing RDBMS with very few work.
- interpreted approach: The resolution tree is explored normally.

Parts of the relations are extracted when needed. This technique is not the most efficient. A compiled technique (with partial evaluation) [EPSI-86], or even better, a half-compiled, half-interpreted technique, could be used and is probably an issue.

- history management allowing to be sure that:
  every extracted tuple is extracted only once from the database.

The advantage of COL is the possibility to treat structured data (imbrication of sets and tuples). In this framework, a COL/RDBMS coupling offers the user a stratum allowing:

- deduction
- data structuration
- programming

Note that with the interpreted approach, the facts that are extracted are surely used, since the instanciation when calling the database is the exact one for the SLD resolution.

Structure of the interface [MOLL-88]

The COL interface is a two module software

- first one is a pre-compiler which collects the conjunctions of edb-predicates in the queues of rules of the user's program, into so called 'edb-lists;. Then for each edb-list, the pre-compiler generates a name and a format for the tuples that will be extracted. Those names are called 'activants'. A part of pre-translation (into SQL) is done at this step. The pre-compiler is integrated into the COL-preformatter.
- second part is a Prolog program defining the edb-call predicate. Its role is to:

• Finish the translation work (of edb-list into SQL). Remember that a part of this job has been processed at pre-compilation time. This work is needed because a part of the conditions (included in the SQL "where" statement) depends on the instanciation of variable at the very last moment before calling the DB.
• Call the database with the corresponding query and load the tuples in a Prolog fact form, into the COL working zone.
• Manage a history of the calls to ensure that no tuple is loaded twice. This work is based on Li's paper [KLM-88].

4-3 Conclusion on the COL interpreter

A prototype has been developed, and works on a SUN 3/50M

Unix machine. The database system is Informix. The Prolog is BIM-Prolog, which is a compiled Prolog. This feature allows reasonable execution times. Note that after pre-compilation, the user's program is compiled by the BIM-Prolog compiler. Then no predicate is dynamic.

The speed of COL can be compared with the speed of a C-Prolog classical interpreter. It is clear that the interpreter is slown down by the COL-unification. A commercial product should be written in a really compiled language like Pascal or C. We are not however convinced that the gain of speed would be superior to 50%, since a great part of the COL-interpreter is the Prolog interpreter itself.

Note that the Prolog programs for COL are very short ones:
COL-interpreter: 5,5 Kbytes
COL-interface: 3,1 Kbytes
of source C-Prolog syntax code including about 20% comments. The BIM compiled code is 9,5 Kbytes long. So COL (with its interface)can be implemented as an extension of any existing Prolog with very few translation work.

As a language for manipulating complex objects, COL is well suited for Prolog programmers, but a final user expects a window-and-icons oriented interface. This issue is being studied.

Performance

By construction:

- the communication between COL and the database is processed via pipes.
- the database is open and closed only once per session.
- during a given session, a given tuple is loaded only once, even if needed many times inside a given resolution or diffent resolutions.
- the pre-compilation is done only once for a given COL program referring to database relations
- the translation COL-SQL is done only for the (where) conditions that are local to a given node of the SLD tree. Everything else is translated only once at precompilation time.

However, due to the absence of Partial Evaluation, some optimizations are ommitted, namely joins that could be processed by the database are sometimes processed by COL.

Of course, to be really fast, such an interpreter should be integrated into a the source of a Prolog interpreter or compiler.

5- Mixing meta-unification with meta-interpretation

Standard meta-interpretation is useful when the resolution strategy has to be controlled or changed, and meta-unification is used when unification itself differs from the standard one. What to do when both of them should be controlled. A simple version of a meta-interpreter-unifier is given here:

Rules must be written under form:

```
   head (Hk, ____):-Q1k , 1. . . , Qnk.

solve (true).
solve ((A,B)):-solve (A),solve (B)
solve (A):-
pre_unification(A,_good_parameters),
    clause  (head(_candidate_literal,
      _good_parameters), B),
    unify  (A,_candidate_literal),
    solve (B).
```

Note that `sl (_)` is no longer necessary. Its role was to intercept classical unification and replace it by the wanted one. This work is now processed by `solve/1` itself. Another feature of `sl/1` was to keep untouched the effect and range of the cut predicate. On the contrary, in the previous program, the cut has to be meta-interpreted. See [SHAP-87] for a survey on such techniques.

The form

`head (Hk,_______):-Q1k , . . . , Qnk.` is not always necessary. The standard form head `Hk:-Q1k , . . . , Qnk` can be enough. Very often, if literal `L` has to be unified with `Hk`, it has same predicate name and arity. Then previous meta-interpreter can be written:

```
solve (true).
solve (A,B)):-solve (A),solve (B).
solve (A):-functor (A,_name,_arity)
      functor (AA,_name,_arity),
      clause (AA,B),
      unify (A,AA),
      solve (B).
```

The predicates `functor (A,_name,_arity), functor (AA, _name, _arity)` are used to build AA, a term with same name and arity as A, but whose parameters are all variables.

Example:

function evaluation can work so

Counter-example:

Immaterial arity unification cannot work with this technique.

Previous program schemas can be modified to really control the resolution strategy, as any meta-interpreter. For example, "right literal first" classical meta-interpreter is:

```
  solve (true).
  solve ((A,B)):-solve (B), solve (A).
  solve (A):-clause (A,B), solve (B).
```

Mixed with meta-unification, it very naturally gives

```
solve (true).
solve ((A,B)):-solve (B), solve (A).
Lsolve (A):- functor (A,_name, _arity),
      functor (AA, _name, _arity),
      clause (AA, B),
      unify (A,AA),
      solve (B).
```

And for any meta-interpretaion feature (such as built-in handling, breath first strategy, partial evaluation . . .).

Conclusion

Meta-unification is a well suited metainterpretation technique to specify and prototype of logic languages where unification has to be controlled. Moreover this technique is fully compatible with standard meta-interpretation. A real prototype, the COL backward chaining interpreter, has been developed with this technique (meta-unification alone), allowing not to rewrite a parser, a resolution tree explorer, a full unifier. The syntax of COL is C-Prolog like, which allows a Prolog programmer to be familiar with COL very quickly. The same remark can be done for any language written by meta-programming technique. No measures have been done yet on the performance of the COL/database interface. Note that this interface can benefit of all existing technologies of coupling Prolog with databases, namely concerning recursion, partial evaluation, syntactic and semantic optimizations and so on. However, the COL interpreter itself (excluding the database problem), which is a meta-interpreter, has not too bad performances (Average 20% slower then a standard Prolog interpreter processing explicitly nesting and unnesting, on program with about 3-level nested relations). This is due to the fact that the control is done only on unification, the rest being handled by the Prolog standard interpreter. Moreover, the COL prototype was developed in BIM-Prolog, which includes a compiler. It seems that Prololg compilers compilates more efficiently meta-unifiers then classical meta-interpreters (the compilation of which does not give good results, due to the clause (_,_) built-in and above all to the call (_) builtin). It seems that this is due to the unification optimizations done at compilation time. Special unification optimizations for meta-unifiers is probably a issue to be studied. Partial evaluation should be taken into account for this purpose.

Bibliography

[ABITE-87] S. Abiteboul S. Grumbach
COL:A logic based language for complex objects
INRIA internal report, august '87.
See also "Une approche de manipulation d'objets complexes" journees BD avancees, port Camargue mai '87

[BERI-86] C. Beeri et al.
Sets and negation in a logical database language (LDL1)
Proc. of ACM PODS 87

[BOCC-86] J.B. Bocca, H.Decker, J.M. Nicolas, L. Vielle, M. Wallace
Some steps towards a DBMS based KBMS.
Information Processing (North Holland)-1986- pp. 1061-1067

[CHAN-78] C.L. Chang
Deduce 2:Further investigations on deduction in relational Data Bases.
In Logic and Data Bases, Plenum Press, 1978, pp. 201-236

[CLAR-78] C. Clark:
Negation as failure in Logic and databases
Edited by Gallaire and Minker, Plenum Press, New York '78

[DELO-82] C. Delobel, M. Adiba
Bases de Donnees et Systemes Relationnels.
Collection Dunod Informatique 449 pages - 1982

[ECRC-86] European Computer-Industry Research Centre:
-J. Bocca: On the Evaluation Stategy of EDUCE.
Proceedings of 1986 ACM-SIGMOD. International Conference on Management of Data. May 1986, Washington, USA
-J. Bocca: EDUCE, a Marriage of convenience: Prolog and a Relational DBMS. Third Symposium on Logic Programming September 86, Salt Lake City, USA

[EPSI-86] P. Coscia, P. Franeschi, J. Kouloumdjian, G. Levi, Li Lei, G.H. Moll, I. Robert, C. Simonell,, L. Torre:
"The Epsilon Knowledge Base Management System: architecture and Database access optimization". Int. Workshop on Integration of Logic and DBs, Venise, December 1986.

[FURA-84] K. Furukawa, A. Takeuchi, S. Kunifuji, H. Yasukawa, M. Ohki and K. Ueda:Mandala: A logic based knowledge programming system. Proc. Int'l Conf. on Fifth Generation Computer Systems (1984), 613-622.

[GALL-81] H. Gallaire
Impacts of Logic on data bases.
Proc. VLDB '81, Cannes 1981, pp. 248-259

[GALL-84] H. Gallaire, J. Minker, J.M. Nicolas
Logic and data bases, a deductive approach.
Computing Survey vol. 1, no2, June 1984, pp. 153-185.

[KLM-87] J. Kouloumdjian, L. Lei, G.H. Moll
Vers une architecture repartie des bases de connaissance
TSI, no 6, vol. 6 '87

[KLM-88] J. Kouloumdjian, L. Lei, G.H. Moll
Prolog-DBMS coupling: a hybrid approach, half interpreted, half compiled. 3rd int. conf. on data and knowledge bases Jerusalem '88

[KOMO-82] J. Komorowski
Partial evaluation as a means for inferencing data structures in an applicative language: a theory and implementation in the case of PROLOG. LINKOPING University-S-581 83-1982

[KOUL-86] J. Kouloumdjian, G.H. Moll et Al.
The EPSILON KBMS: Architecture and data base access optimization in /WILD-86/

[LIJK-86] L. Lei, J. Kouloumdjian
An implementation of a partial evaluation system.
Internal report - University of Lyon (1986).

[LYON-86] J. Kouloumdjian, I. de St. Victor, L. Lei, S. Djennaoui, G.H Moll
The Communication Processor. University of Lyon
Epsilon 2nd year report P530-15, December '86

[MOLL-88] G.-H. Moll
A complex object logical language (COL) interpreter for relational databases. Journees bases de donnees deductives et bases de donnees orientees objets. Paris, Decembre 1988

[NICO-83] J.M. Nicolas et K. Yazdanian
An outline of BDGEN: A deductive DBMS.
PROC. IFIP 83 Congress North Holland 1983 pp. 711-717

[VASS-84] Y. Vassiliou, J. Clifford, M. Jarke
Access to specific declarative knowledge by expert systems: the impact of logic programming
Decision Support Systems-Vol. 1 - No. 1 - 1984

[VLenk-84] R. Venken
A PROLOG meta-interpreter for partial evaluation and its application to source to source transformation and query optimizations ECAI- 1984

[VILD-86] Workshop on the integration of logic programming and data bases. VENISE-Dec 1986

Book Reviews

Readings in distributed artificial intelligence

Alan H. Bond and Les Gasser (editors)
[California Institute of Technology;
University of Southern California]

San Mateo, CA: Morgan Kaufmann, 1988, xvii+649 pp
(Distributed in Canada by John Wiley and Sons Canada Ltd)
Paperback, ISBN 0-934613-63-X, cdn$42.50

Reviewed by
Innes A. Ferguson
University of Cambridge

In what has almost become a tradition, the publishers of this book, Morgan Kaufmann, have released their newest title in their Readings series. This one reviews the ever-growing field of distributed artificial intelligence (DAI), which in the editors' words is defined as "the subfield of AI concerned with concurrency in AI computations, at many levels". The aim of the book is to provide the reader with a core set of papers from the field, as well as an overview of the basic issues and techniques pertaining to DAI. The book is aimed at students (primarily at the graduate level), practitioners, and researchers in the field, and the basic requirements of the reader include some knowledge of AI, logic, and discrete mathematics (although these prerequisites vary from paper to paper). It is worth mentioning that while research termed 'connectionist' or 'neural' in the past fell under the umbrella of DAI, this is no longer the case; consequently none of the papers included in this collection deal with what is perhaps more appropriately termed 'parallel AI'.

The book is divided into three parts. Part I, Orientation, comprises two chapters. The first of these provides the DAI newcomer with a general and up-to-date analysis of the subject, including its definition and scope, historical roots (from early work on perceptrons through to 'futuristic' agoric open systems), its rationale and aims, the basic issues involved, and – for the eager graduate student – a list of several research problems yet to be solved. The second chapter includes a comprehensive, 300-entry, subject-indexed bibliography of DAI research and 'related' fields (e.g. AI, economics, game theory, philosophy, social science, and organization theory).

Part II, Basic DAI Problems and Approaches, comprises chapters 3 through 5 and includes papers covering most of the fundamental aspects of DAI. Chapter 3 focuses on task allocation and distribution, and covers such topics as distributed control, cooperative architectures, distributed situation assessment and interpretation, coordination in organizations and markets, and distributed planning. Chapter 4 looks at agent coherence and coordination, addressing such topics as cooperating agents, speech act planning, multi-agent planning and communication, multi-agent conflict, distributed problem solving, and open systems. Finally, chapter 5 looks at interaction languages and protocols, and includes, among others, papers on actor systems and negotiation mechanisms.

Part III, Implementation Frameworks and DAI Applications (chapters 6 through 8), includes papers dealing with the more low-level and practical problems in DAI. Chapter 6 reviews several implementation languages and systems, and describes work using a concurrent object-oriented paradigm, agoric architectures, and fully-blown, specialized DAI environments. Chapter 7 looks at a particularly successful DAI programming/control structure, the blackboard, while Chapter 8 highlights several applications of DAI techniques in such domains as document retrieval, office systems modelling, and air traffic control, to name a few.

The total number of papers included in this book is 47, about two-thirds of which are grouped under Part II. The emphasis of the book, then, is on fundamental issues and problems in DAI, rather than on applications or programming environments. At the time of print, the editors claimed that the published literature in DAI exceeded 500 papers. However accurate this claim might be, it is reasonable to rate their bibliography as the most extensive and useful to date. (I should mention also that I found the book's index very useful for cross-referencing on subjects, authors, and keywords.)

Besides providing a well-balanced and representative collection of papers from the field of DAI, this book presents the most extensive and up-to-date introduction to the subject so far. (The previous 'best' appeared in J.S. Rosenschein's 1985 Stanford Ph.D. thesis *Rational interaction: Cooperation among intelligent agents.*) Its bibliography is excellent and will certainly help the DAI aficionado minimize his/her effort in tracking down references. From my own experience, I found that most of the papers required for my own research (and published prior to AAAI-88), were either included in the book itself or else listed in the bibliography. This well-edited book strikes a good balance between theoretical and practical issues, and is an excellent way to get to know who's who and what's what in DAI and related fields.

Innes Ferguson is a Ph.D. candidate in the Computer Laboratory at the University of Cambridge (England), and is currently working on aspects of rational, multi-agent planning in open worlds. His primary funding is provided through a Postgraduate Scholarship from Bell-Northern Research Ltd., Ottawa, Ont.

Outils logiques pour le traitement du temps:
De la linguistique à l'intelligence artificielle

Hélène Bestougeff et Gérard Ligozat
[Université Paris 7 et CNRS]

Paris: Masson, 1989, 272 p.
(études et recherches en informatique)
Couverture rigide, ISBN 2-225-81632-8

Compte rendu par
Nathalie Japkowicz,
University of Toronto

Basé sur une critique des diverses approches à l'étude du temps qui, selon les auteurs souffre de l'absence d'un cadre général, ce livre, *Outils logiques pour le traitement du temps: de la linguistique à l'intelligence artificielle* se propose de regrouper les résultats solides et déjà bien établis du domaine. Les résultats exposés par cet ouvrage se rattachent à des disciplines diverses telle que la linguistique, la logique et l'intelligence artificielle. Une étude systématique des structures et des logiques temporelles est tout d'abord conduite, suivie de la déscription de quelques systèmes appliqués. Ce livre est déstiné à des étudiants en informatique et linguistique. Il a pour but de permettre au lecteur de se familiariser avec les outils liés à la logique temporelle, et d'aborder directement la recherche dans le domaine du traitement du temps.

L'ouvrage se compose de quatres chapitres et d'une annexe.

Le premier chapitre est une introduction à la problématique du traitement du temps. Le problème est présenté sous la forme d'un schéma tripolaire dont les pôles sont: le *système de représentation*, le *language naturel* et le *monde réel*. Trois des relations entre ces pôles représentent les points principaux qu'une étude sur le temps doit prendre en consideration. Cinq classes de notions temporelles sont alors introduites. Chacune de ces classes est illustrée par de nombreux examples. Cette discussion initie le lecteur à la complexité du traitement du temps qui transcende la simple notion de précédence temporelle. Dans un dernier temps, enfin, sont introduites les préoccupations de l'intelligence artificielle quant au traitement du temps. Quelques exemples de systèmes existant sont mentionnés ainsi que leur limitations.

Les deux chapitres suivants constituent le noyeau technique de l'ouvrage. Ils sont tous deux très abstraits. Les discussions sont menées de manière formelle. Pour illustrer les concepts nouveaux, des examples sont donnés qui aident à la compréhension. Néanmoins, certaines idées restent difficiles d'accès.

Le second chapitre se consacre aux stuctures temporelles. Le cas ponctuel et le cas étendu (le cas des structures d'intervalles) sont tous deux considérés ainsi que le passage d'un point de vue a l'autre. Dans un premier temps, il est rappelé au lecteur qu'un système de représentation du temps se base sur trois choix fondamentaux: le choix des *entités* temporelles primitives, le choix des *relations* entre ces entités et le choix des *propriétés* que doivent satisfaire la (ou les) relation(s). Une fois ce fait établi, les structures ponctuelles sont tout d'abord examinées. La notion de *cadre temporel* est introduite et les définitions et propriétés des relations binaires rappelées. Une bonne partie de la section est consacrée à l'étude des propriétés des cadres temporels ainsi qu'à leur interprétation en termes de linguistique temporelle. Les structures d'intervalles sont ensuite examinées. Les grandes lignes de la théorie sont dégagées mais pas de manière aussi systématique que dans le cas ponctuel. La section suivante est consacrée aux systèmes d'Allen et à la notion d'*intervalles généralisés* qui permettent de modeliser des phénomènes ayant plusieurs phases dans leur déroulement. Cette section est bien détaillée. Dans un dernier temps, le passage du point de vue ponctuel au point de vue des intervalles et l'inverse—c'est à dire les techniques de construction d'un système d'intervalles à partir d'un ensemble de points munis d'une relation d'ordre et le passage des intervalles à "leurs points sous-jacents"—sont étudiés. Deux cas sont envisagés: le premier, un cas général faisant appel au notions de *filtre* et d'*ultrafiltre*; le second, un cas particulier s'appliquant aux systèmes d'Allen.

Le troisième chapitre aborde le problème du traitement du temps du point de vue du language logique. Le but principal du chapitre est de présenter de manière systématique plusieurs systèmes logiques associés aux différentes propriétés rencontrées dans les chapitres précédents. Il étudie en particulier les rapports entre les propriétés des structures et les systèmes axiomatiques, et en tire des résultats de complétude, et plus généralement de correspondances.

Les éléments de base de la logique de Prior sont tout d'abord décrits puis developpés de fa\con plus poussée après l'introduction de nouvelles données concernant les cadres temporels. Les résultats de complétude entre l'axiomatique ainsi développée et la classe de tous les cadres temporels sont alors démontrés, utilisant la méthode de Henkin. D'autres résultats de complétudes sont ensuite présentés. La section suivante introduit deux techniques plus particulières permettant de prouver des résultats en logique temporelle: la *méthode des filtration* et la *méthode de traduction* d'une logique temporelle en des résultats connus de la logique de second ordre. Ces deux méthodes sont utilisées pour prouver la décidabilité de certaines logiques temporelles. Une *théorie des correspondances* entre les propriétés des cadres temporels et les axiomes logiques est alors construite utilisant la méthode de traduction. Dans une dernière section, quelques extensions des logiques prioriennes sont discutées. Dans le cadre des logiques temporelles basées sur les points, la logique en S ("Since") et U ("Until") de Kamp et la logique de Prior étendue par l'opérateur J ("maintenant") sont toutes deux presentées. Un résumé des principaux resultats concernant les fragments modaux des logiques temporelles prioriennes est également proposé. Dans le cadre des logiques temporelles basées sur les intervalles, quelques résultats concernant la logique de Van Benthem sont revus très rapidement, puis enfin, le système de Halpern et Shoham qui est le pendant des systemes d'Allen du point de vue logique est presenté en détail. Dans un dernier temps, certains problèmes causés par le passage d'une logique propositionnelle à une logique temporelle du premier ordre sont informellement discutés.

Le quatrième chapitre fait le lien entre les théories developpées dans les deux chapitres précédents et leurs applications. Plusieurs systèmes considérés comme typiques du domaine sont presentés. Le chapitre se divise en trois parties. La premiere partie décrit la *logique intensionnelle de Montague* et son extension par Dowty. Cette partie décrit la théorie puis illustre la façon dont cette théorie peut servir à traduire des expressions temporelles du language naturel dans le language de la logique intensionnelle. La seconde partie présente trois systèmes considérés comme représentatifs de la démarche du raisonnement temporel en intelligence artificielle. Le point important dans ce contexte est de choisir des structures temporelles qui correspondent aux objectifs à atteindre. Le premier système étudié est le système d'Allen. Le second est celui de McDermott. L'objectif principal de ces deux systèmes est la planification. Le troisième système est celui de Kowalski et Sergot dont l'objectif est la manipulation de bases de données. La troisième partie étudie deux types de procédures particulières nécéssaires pour la mise en oeuvre informatique d'un système de gestion d'informations temporelles: le *test de cohérence* et les *procédures de décision*. Le test de cohérence est étudié dans le contexte des relations établies par Allen. Dans le cadre des procédures de décision, deux approches sont discutées: la méthode des tableaux semantiques et la méthode de résolution non-clausale dans les logiques temporelles.

L'annexe introduit les notions essentielles de la logique classique et les notations utilisées au cours de l'ouvrage.

Ainsi que son titre l'indique, ce livre est une collection d' "outils" pour le traitement du temps. Théories et techniques sont bien exposées, et de manière consistente mais seulement succintement analysées. Ce livre est avancé dans le sens où le lecteur en tirera profit dans la mesure où il a déjà étudié des systèmes de gestion du temps, s'interesse à une formalisation de ces systèmes et est capable d'en faire sa propre analyse.

Un autre point à considerer est l'orientation de cet ouvrage. Tout au cours de ce livre, le problème du traitement du temps est envisagé de manière mathématique plutôt qu' informatique. Les auteurs se soucient plus de la formalisation des théories que de l'interêt d'une technique par rapport à une autre dans le cadre pratique. Le quatrième chapitre aborde le sujet mais de manière superficielle: encore une fois, les systèmes sont décrit mais pas analysés en détail.

En définitive, ce livre, comme promis, construit un cadre général assez complet, nécéssaire à l'étude du temps. Cependant, étant donnée l'ampleur du sujet, certains aspects ont été négligés. A mon avis, les auteurs n'analysent pas assez les techniques et systèmes qu'ils décrivent et également ne les replacent pas de manière systématique dans leur contexte informatique. Dans l'ensemble, cependant cet ouvrage présente une bonne introduction aux differente structures et logiques temporelles qui peuvent être utiles à un informaticien ou un linguiste interéssé par l'étude du traitement du temps.

Nathalie Japkowicz est une candidate à la maîtrise dans le groupe de Linguistique Computationnelle de l'Université de Toronto. Sa recherche l'a conduite à étudier certains aspects de la logique temporelle.

Readings in artificial intelligence and databases

John Mylopoulos and Michael L. Brodie (editors)
[University of Toronto and GTE Laboratories]

San Mateo, CA: Morgan Kaufmann Publishers, 1989, vii+688 pp (Distributed in Canada by John Wiley and Sons Canada Ltd) Paperbound, ISBN 0-934613-53-2, cdn$42.50

Reviewed by
Stephen Regoczei,
Trent University

The book under review is a welcome contribution to the effort of bringing the fields of AI and databases closer together – for the benefit of both. The book succeeds in what seems to be a hopeless task. The two fields are so disparate, they are such different undertakings, that, on the face of it, there seems to be no way that they could be shown to interact.

Databases, and the guiding concerns of database work, grew out of business data processing. The concept of database was founded on metaphors such as the record, the business report, file folders and filing cabinets. The report was pictured as a tabular presentation of information, with column headings, and typically fitting onto a single sheet of paper. Database thinking was strongly influenced by the archetypal I/O medium: the punched card. Iteration was the dominant control structure in processing the data. More recently, the metaphors changed somewhat. The dominant interface is now the business form, which is to be "filled in". Processing is now on-line. The term "database" is now a misnomer: "records base", or "relations base" would be more accurate. With business forms being the guiding metaphor of database use, we are now looking at databases as permanent storage not only for records, but for objects of a more general nature. Nevertheless, one thing did not change: a strict conceptual separation between a process and the material upon which it works. The separation of code from data provides an environment that is relatively uncomplicated.

In artificial intelligence, the basic concerns could not have been more different. AI, to say the least, was not based on the business environment. Problem solving, game playing, symbolic programming, recursion, and functions producing results with no permanent record or audit trail being kept – these are the main features of the AI landscape. There is no strict separation or distinction between code and data. Code works upon itself and transforms itself as execution proceeds. This is enshrined in the early metaphor of AI: thinking as list processing.

Yet, somehow, the two fields approach each other, perhaps to form a third area of activity. Expert systems need vast amounts of data-like material to perform realistically. Databases need intelligent management of constraints and update rules to overcome the severe shortcomings in preserving database integrity.

Although AI and DB concerns seem so far removed from each other even today, the book contains a paper by Roussopoulos and Mylopoulos that, as early as 1975, suggested the use of semantic networks for the management of databases.

The book brings together the best papers that describe both actual and potential interactions in the two fields. It opens with a good introduction and a historical outline, as well as a survey of database management by Brodie and Manola. It closes with a prediction by Brodie about how the AI and DB technologies will be working together in the future. As for the assessment of the current state of affairs, the editors state that "the reader should not expect to find a deep integration of . . . AI and database concepts . . . Although there has been some progress, significant integration has yet to occur."

The progress is well documented through papers by Bachman, Smith and Smith, Kent, Levesque, and Reiter, amongst others. The 39 papers reprinted in the book give a convincing demonstration that cooperation between AI and DB is, in fact, possible.

There are excellent papers by the editors to describe the history of the AI-DB interaction. Illuminating commentaries for each section of the book provide guidance to the reader in contextualizing the reprinted papers.

The book does not cover the entire range of AI-DB. The editors clearly state that they are not covering areas such as natural-language front-ends to databases, or expert-system applications in improving the design and performance of databases. Yet, in addition to the admitted limitations, there is an interesting tacit omission. There are no papers in knowledge acquisition on the AI side, and data modelling on the DB side. This omission is interesting because of the parallels. Knowledge acquisition is to knowledge representation as data modelling is to data models. There are papers in the collection both on knowledge representation and on data models, but there are no papers covering the activities of knowledge acquisition and data modelling.

In conclusion, the book is an excellent document proving that the seemingly impossible is not impossible at all. It is a valuable contribution to the literature and indispensable to those who are interested in AI-DB interaction.

Stephen Regoczei is an associate professor of Computer Science at Trent University, Peterborough, Ontario. His research concerns knowledge acquisition methodologies. Since he wrote this review, he has joined John Mylopoulos's knowledge-base management systems project at the University of Toronto.

Briefly Noted

Perspectives in artificial intelligence
Volume 1: Expert systems:
Applications and technical foundations
Volume 2: Machine translation, NLP, databases and computer-aided instruction

John A. Campbell and José Cuena (editors)
[University College London and Universidad Politécnica de Madrid]

Chichester, England: Ellis Horwood, 1989, Vol. 1, 162pp; Vol. 2, 211 pp
(Ellis Horwood series in artificial intelligence)
(Distributed in Canada by John Wiley and Sons Canada Ltd)
Vol 1, Hardbound, ISBN 0-7458-0659-7 and 0-470-21434-1, cdn$77.95
Vol 2, Hardbound, ISBN 0-7458-0660-0 and 0-470-21435-X, cdn$90.95

These volumes collect the invited papers given at a conference on AI held in association with the Second World Basque Congress. Some papers are overviews of AI subfields, while others report the authors' own research. The result is rather a mixed bag, as the subtitles of the volumes suggest. Despite the Basque context, most of the papers have no particular Basque connection, and the set may be thought of as a general AI collection. Publication as two small volumes (one with the expert systems papers, the other with everything else) instead of one largish one seems to serve no purpose other than to increase the total price.—*G.H.*

Sur les fondements de la mathématique: Fragments (Discussions préalables, méréologie, ontologie) (On the foundations of mathematics: Fragments: Preliminary discussions, mereology, and ontology)

Stanislaw Lésniewski
[1886–1939]

(Traduit du polonais par / Translated from Polish by Georges Kalinowski)
Paris: Hermes, 1989, 148 pages
(Langue, raisonnement, calcul)
Livre relié, ISBN 2-86601-178-3, FF395

L'intérêt de ce livre pour les chercheurs en intelligence artificielle tient surtout à son exposition d'une théorie de la méréologie (étude des relations entre les touts et les parties), qui apparaît maintenant être une partie importante du raisonnement intuitif (une partie importante du raisonnement dit "courant") comporte intuitivement un raisonnement sur les ingrédients et composantes des corps. Le livre traite également d'ontologie (choix des individus), autre sujet à la mode.—*J.F.L.*

AI researchers will find in this book an exposition of mereology (a general theory of part-whole relationships). This topic is now increasingly perceived as an important component of "common-sense" reasoning dealing with ingredients and properties dependending on composition of bodies. The book also covers formal ontology (choice of individuals), another common topic in AI.—*J.F.L.*

Dialogue oral homme–machine: Connaisances linguistiques stratégies et architectures des systèmes (Spoken human–machine dialogue: Linguistic knowledge strategies and system architectures)

Jean-Marie Pierrel
[Université de Nancy I]

Paris: Hermes, 1987, 239 pages
(Langue, raisonnement, calcul)
Livre relié, ISBN 2-86601-105-8, FF195

Ce livre présente un survol didactique du problème de la compréhension automatique de la parole. Le livre est structuré en trois parties (motivation et contexte, reconnaissance et compréhension de phrases, compréhension et gestion de dialogues). Aucune connaissance préalable du domaine ne semble supposée, quoiqu'un connaissance de base de l'informatique et de l'intelligence artificielle est probablement souhaitable.—*J.F.L.*

This French textbook presents a survey of speech understanding research. The book has three parts (motivation and context, recognition and sentence understanding, dialogue management and understanding). No prior knowledge of the area is assumed, though a basic knowledge of computer science and artificial intelligence techniques will likely be useful to understanding the subject matter.—*J.F.L.*

Computers and thought: A practical introduction to artificial intelligence

Mike Sharples, David Hogg, Chris Hutchison, Steve Torrance, and David Young
[University of Sussex]

Cambridge, MA: The MIT Press, 1989, xxix+401 pp
(Explorations in cognitive science series)
Hardbound, ISBN 0-262-19285-3, us$25.00

Computers and thought provides a unified, self-contained introduction to artificial intelligence for readers with little or no computing background. It presents an original extended AI programming project—the Automated Tourist Guide exercise—throughout the main chapters of the text to illustrate the material covered and to show how AI actually works.—*From the publisher's announcement.*

Books Received

Books listed below that are marked with a + will be reviewed in a future issue. Reviewers are still sought for those marked with a *.

Readers who wish to review books for *Canadian AI* should write, outlining their qualifications, to the book review editor, Graeme Hirst, Department of Computer Science, University of Toronto, Toronto, Canada M5S 1A4. Obviously, we cannot promise the availability of books in anyone's exact area of interest.

Authors and publishers who wish their books to be considered for review in *Canadian AI* should send a copy to the book review editor at the address above. All books received will be listed, but not all can be reviewed.

+Deduction systems in artificial intelligence
Karl Hans Bläsius and Hans-Jürgen Bürckert (editors)
[IBM Stuttgart and Universität Kaiserslautern]
Chichester, England: Ellis Horwood, 1989, 238 pp
(Ellis Horwood series in artificial intelligence)
(Distributed in Canada by John Wiley and Sons Canada Ltd)
Hardbound, ISBN 0-7458-0409-8 and 0-470-21550-X, cdn$103.95

Situations, language and logic
Jens Erik Fenstad, Per-Kristian Halvorsen, Tore Langholm, and Johan van Benthem
[University of Oslo, Xerox Palo Alto Research Center, Stanford University, and University of Amsterdam]
Dordrecht: Reidel, 1987, viii+186 pp
(Studies in linguistics and philosophy 34)
Hardbound, ISBN 1-55608-048-4, Dfl 125.00, \L38.75

+L'inference en langue naturel: Le problème des connecteurs; Représentation et calcul
(Inference in natural language: The problem of the connectors; Representation and computation)
Jacques Jayez
[Ecole des hautes études en sciences sociales]
Paris: Hermes, 1988, 287 pages (Langue, raisonnement, calcul) Livre relié, ISBN 2-86601-118-X, FF195

Frequency analysis of English vocabulary and grammar, based on the LOB corpus.
Volume 1: Tag frequencies and word frequencies.
Volume 2: Tag combinations and word combinations.
Stig Johansson and Knut Hofland
[University of Oslo and Norwegian Computing Centre for the Humanities]
Oxford: Clarendon Press, 1989, Vol 1: vii+400 pp, Vol 2: v+380 pp Hardbound, Vol 1: ISBN 0-19-824221-2, Vol 2: 0-19-824222-0, cdn$110.00 each

Artificial intelligence techniques in language learning
Rex W. Last
[Department of Modern Languages, University of Dundee]
Chichester, England: Ellis Horwood, 1989, 173 pp
(Ellis Horwood series in computers and their applications)
(Distributed in Canada by John Wiley and Sons Canada Ltd)
Hardbound, ISBN 0-7458-0177-3 and 0-470-21503-8, cdn$77.95

+Modélisation du dialogue: Représentation de l'inférence argumentative
(Modelling dialogue: Representation of inferential argumentation)
Jacques Moeschler
[Université de Genève]
Paris: Hermes, 1989, 266 pages
(Langue, raisonnement, calcul)
Livre relié, ISBN 2-86601-191-0, FF200

Natural language processing technologies in artificial intelligence: The science and industry perspective
Klaus K. Obermeier
[Battelle Laboratories]
Chichester, England: Ellis Horwood, 1989, 263 pp
(Ellis Horwood series in artificial intelligence)
(Distributed in Canada by John Wiley and Sons Canada Ltd)
Hardbound, ISBN 0-7458-0562-0 and 0-470-21528-3, cdn$77.95

Recent developments and applications of natural language processing
Jeremy Peckham (editor)
[Logica]
London: Kogan Page, 1989, xiv+272 pp.
(UNICOM applied technology reports)
Hardbound, ISBN 1-85091-682-9, \L50.00

Temporal representation and inference
Barry Richards, Unge Bethke, Jaap van der Does, Jon Oberlander
[Imperial College, University of Edinburgh, University of Amsterdam, and University of Edinburgh]
London: Academic Press, 1989, xxi+367 pp
Hardbound, ISBN 0-12-587770-6, \L19.50

***Inside case-based reasoning**
Christopher K. Riesbeck and Roger C. Schank
[Northwestern University]
Hillsdale, NJ: Lawrence Erlbaum Associates, 1989, xxii+423 pp Hardbound, ISBN 0-89859-767-6, us$34.50

+L'intelligence artificielle et le langage.
Volume 1: Représentations des connaissances;
Volume 2: Processus de compréhension
(Artificial intelligence and language.
Volume 1: Knowledge representation;
Volume 2: Processes of comprehension)
Gérard Sabah
[CNRS]
Paris: Hermes, Vol. 1: 1988, 352 pages, Vol 2: 1989, 411 pages (Langue, raisonnement, calcul) Livre relié, Vol 1: ISBN 2-86601-134-1, Vol 2: ISBN 2-86601-187-2, FF240 chaque

+Raisonnements sur des informations incomplètes en intelligence artificielle: Comparison de formailismes à partir d'un exemple
(Reasoning about incomplete informations in artificial

intelligence: A comparison of formalisms on an example)
Léa Sombé (pseud.)
Toulouse: Teknea, 1989, 221 pages
Livre broché, ISBN 2-87717-010-1, FF150

Scheme and the art of programming
George Springer and Daniel P. Friedman
[Indiana University]
Cambridge, MA: The MIT Press and New York: McGraw-Hill, 1989, xxiv+595 pp.
(The MIT electrical engineering and computer science series) Hardbound, ISBN 0-262-19288-8, $39.95

Common Lisp:
A gentle introduction to symbolic computation
David S. Touretzky
[Carnegie Mellon University]
Redwood City, CA: Benjamin / Cummings, 1990, xxiii+564 pp
Paperbound, ISBN 0-8053-0492-4, $27.95

Technical Report

SFB 314 - Special Collaborative Program on Artificial Intelligence and Knowledge-Based Systems
AI - Laboratory at Department of Computer Science
Director: Prof. Dr. W. Wahlster
University of Saarbruecken, FB 10 Informatik IV, Im Stadwald 15, D-6600 Saarbruecken 11, Fed. Rep. of Germany
Ph: 0681/302-3263

New Reports and Memos

(# ::= out of print)
Single copies can be ordered from Doris Borchers under the above address.

Report No. 29
Schmauks, D., N. Reithinger
Generating Multimodal Output - Conditions, Advantages and Problems. January 1988.
In: COLING-88, Budapest 1988. Proceedings. 584-588.

Abstract: In natural communication situations, multimodal referent specification is frequent and efficient. The linguistic component are deictic expressions, e.g. 'this' and 'here'. Extralinguistic devices in dialogs are different body movements, mainly pointing gestures. Their functional equivalent in texts are means like arrows and indices. This paper has two intentions. First, it discusses the advantages of multimodal reference in interhuman communication, which motivate the integration of extralinguistic "pointing" devices into NL dialog systems. The generation of multimodal output poses specific problems, which have no counterpart in the analysis of multimodal input. The second part presents the strategy for generating multimodal output which has been developed within the framework of the XTRA system (a NL access system to expert systems). XTRA allows the combination of verbal descriptions and pointing gestures in order to specify elements of the given visual context, i.e. a form displayed on the screen. The component POPEL generates referential expressions which may be accompanied by a pointing gesture. The appearance of these gestures depends on several factors, e.g. the type of referent (whether it is a region or an entry of the form) and its complexity.

Report No. 30
Wahlster, W., A. Kobsa
User Models in Dialog Systems. January 1988.
To appear in: Kobsa, A.,W. Wahlster (eds.): User Models in Dialog Systems. Berlin etc.: Springer, 1988. (Extended and revised version of Report No. 3).

Abstract: This paper surveys the field of user modeling in artificial intelligence dialog systems. First, reasons why user modeling has become so important in the last few years are pointed out, and definitions are proposed for the terms 'user model' and 'user modeling component'. Research within and outside of artificial intelligence which is related to user modeling in dialog systems is discussed. In section two, techniques for constructing user models in the course of a dialog are presented and, in section three, recent proposals for representing a wide range of assumptions about a user's beliefs and goals in a system's knowledge base are surveyed. Examples for the application of user models in systems developed to date are then given, and some social implications discussed. Finally, unsolved problems like coping with collective beliefs or resource-limited processes are investigated, and prospects for application-oriented research are outlined. Although the survey is restricted to user models in natural-language dialog systems, most of the concepts and methods discussed can be extended to AI dialog systems in general.

Report No. 31
André, E., G. Herzog, T. Rist
On the Simultaneous Interpretation and Natural Language Description of Real World Image Sequences: The System SOCCER. April 1988.
In: ECAI-88. Proceedings. London: Pitman, 1988, 449-454.

Abstract: The aim of previous attempts at connecting vision systems and natural language systems has been to provide a retrospective description of the analysed image sequence. The step from such an a posteriori approach towards simultaneous natural language description reveals a problem which has not yet been dealt with in generation systems. Automatic generation of simultaneous descriptions calls for the application of an incremental event recognition strategy and for the adequate coordination of event recognition and language production. In order to enable free interaction between these processes, it is useful to implement them in parallel. In this paper the system SOCCER will be presented, which is based upon such a conception. Short sections of soccer games have been chosen as the domain of discourse. In analogy to radio reports, the system generates a description of the game which it is watching and which the listener cannot see.

Report No. 32
Kemke, C.
Der Neuere Konnektionismus. Ein Überlick. May 1988.
In: Informatik-Specktrum, 11.3, Juni 1988, 143-162.

Abstract: "Connectionism" denotes a new multi-disciplinary research paradigm concerned with the analysis and construction of aggregated systems which model main features and characteristics of neural networks. Connectionist systems serve as models of parallel distributed information processing in cognitive systems. This paper surveys the foundations and applications of connectionist models.

Report No. 33
Retz-Schmidt, G.
Various Views on Spatial Prepositions. May 1988.
In: AI Magazine, 9.2, 1988, 95-105.

Abstract: In this article, principles involving the intrinsic, deictic, and extrinsic use of spatial prepositions are examined from the viewpoint of linguistic, psychological, and Artificial Intelligence approaches. First, some important notions are defined. After that, those prepositions that permit intrinsic, deictic, and extrinsic use are specified. Next, we examine for all three cases how the frame of reference is determined. Then, we look at ambiguities in the use of prepositions and how they can be resolved. Finally, we introduce the natural language dialog system CITYTOUR, that can cope with the intrinsic, deictic, and extrinsic use of spatial prepositions, and compare it to the approaches dealt with in the previous sections as well as to some other AI systems.

Report No. 34
Ripplinger, B., A. Kobsa
PLUG: Benutzerführung auf Basis einer dynamisch veränderlichen Zielhierarchie. May 1988.
In: Hoeppner, W. (ed): Künstliche Intelligenz. GWAI-88. 12th German Workshop on Artificial Intelligence. Proceedings. Berlin etc.: Springer, 1988, 236-245.

Report No. 35
Schirra, J.R.J.
Deklarative Programme in einem Aktor-System: MEGA-ACT. May 1988.
In: KI, 2.3/2.4, 1988, 4-9/4-12.

Abstract: The construction of declarative programs is studied by means of the actor system MEGA-ACT. Starting point of this investigation is the problem to realize the essentially parallel actor systems by means of "normal" computers with only one processor. The well-known sequentialization of the simultaniously active actors by means of messages is too inflexible and leads easily to inconsistencies. Therefore, the decision of which actor be active for which period of time shall be made by the system itself. The programs must be represented declaratively, i.e. their structure must be accessible for the system at least in some parts. Thus, a general scheme for declarative programs is presented as the "core" of the system MEGA-ACT which is implemented in the Flavor-like programming language FRL. This scheme allows for pseudo-parallel interpretations of the programs - interpretations which are independent of the sequentialization of FRL by means of messages.

Report No. 36
Retz-Schmidt, G.
A REPLAI of SOCCER: Recognizing Intentions in the Domain of Soccer Games. June 1988.
In: ECAI-88. Proceedings. London: Pitman, 1988, 455-457.

Abstract: This paper deals with the problem of the recognition and verbalization of intentions in the context of systems for the natural language description of image sequences. First of all, some motivation is given as to the importance of verbalizing intentions in the natural language description of image sequences. Next, we try to give a closer account of what intentions are and how they can be recognized. After that, we describe the system REPLAI, that recognizes intentions in the domain of soccer games. Finally, we refer to the future directions of our work.

Report No. 37
Wahlster, W.
User and Discourse Models for Multimodal Communication. June 1988.
To appear in: Sullivan, J.W., S.W. Tyler (eds.): Architectures for Intelligent Interfaces: Elements and Prototypes. Reading: Addison-Wesley, 1988.

Abstract: In face-to-face conversation humans frequently use deictic gestures parallel to verbal descriptions for referent identification. Such a multimodal mode of communication is of great importance for intelligent interfaces, as it simplifies and speeds up reference to objects in a visualized application domain. Natural pointing behavior is very flexible, but also possibly ambiguous or vague, so that without a careful analysis of the discourse context of a gesture there would be a high risk of reference failure. The subject of this paper is how the user and discourse model of an intelligent interface influences the comprehension and production of natural language with coordinated pointing, and conversely how multimodal communication influences the user and discourse model. After a brief description the deixis analyzer of our XTRA system, which handles a variety of tactile gestures, including different granularities, inexact pointing gestures and pars-pro-toto deixis, we present some empirical results of an experiment, which investigates the similarities and differences between natural pointing in face-to-face communication and simulated pointing using our system. This paper focuses on consequences of these investigations for our present work on an extended version of the deixis analyzer and a gesture generator currently under development. We show how gestures can be used to shift focus and how focus can be used to disambiguate gestures. Finally, we discuss the impact of the user model on the decision of the presentation planning component, as to whether to use a pointing gesture, a verbal description, or both, for referent identification.

Report No. 38
Harbusch, K.
Effiziente Analyse natürlicher Sprache mit TAGs.
June 1988. In: Batori, I.S., U. Hahn, M. Pinkal, W. Wahlster (eds.): Computerlinguistik und ihre theoretischen Grundlagen. Symposium, Saabrücken, März 1988. Proceedings. Berlin etc.: Springer, 1988, 79-103.

Abstract: Natural language parsing places three main demands

upon a grammar formalism. Its rules should be easy to write and comprehend, the formalism should be powerful enough to enable the encoding of diverse linguistic problems, and the parsing process should be efficient. I will now introduce a formalism which meets all of the requirements mentioned above - the Tree Adjoining Grammar (TAG). Here, simple structures of sentences are written as trees, each of which being able to be combined. TAGs are more powerful than context-free languages - belonging to the class of 'mildy context-sensitive Grammars'. Given that n is the length of the input sentence, the time complexity of the parsing algorithm is $O(n^4)$. This article contains a theoretical and a practical section. In the former, a description of the formalism is given and, in using a well known context-sensitive grammar, its power is demonstrated. Primary emphasis though is placed upon describing my parsing algorithm for TAGs with time and space complexity of $O(n^4)$ because, up to now, only one requiring time complexity of $O(n^6)$ was known. In the following section, the linguistic adequacy of the formalism is confirmed by defining a restricted German grammar for some linguistic phenomena such as, for example, verb raising.

Report No. 39
Schifferer, K.
TAGDevEnv. Eine Werkbank für TAGs. June 1988.
In: Batori, I.S., U.Hahn, M. Pinkal, W. Wahlster (eds.): Computerlinguistic und ihre theoretischen Grundlagen. Symposium, Saarbrücken, März 1988. Proceedings. Berlin etc.: Springer, 1988, 152-171.

Abstract: Intelligent natural language processsing is based upon the systematic recording of linguistic knowledge. This knowledge, in turn, has to be formalized in order to meet the requirements of natural language processing systems. Also, knowledge transfer between linguistics and these NL systems should be direct - bypassing, if possible, specialist in the field of knowledge representation techniques. The following paper outlines the TAGDevEnv - an interactive development environment for Tree Adjoining Grammars (TAGs). TAG is a grammar formalism for the formal description of linguistic knowledge. The TAGDevEnv offers language experts a comfortable and efficient working environment containing various tools for the construction of linguistic knowledge bases. Apart from a graphically oriented structure editor for TAGs, it also includes test facilities for maintaining the consistency of the developed grammars, interfaces for lexicon construction and maintenance, as well as a TAG parser for syntactic analysis of natural language sentences.

Report No. 40
Finkler, W., G. Neumann
MORPHIX. A Fast Realization of a Classification-Based Approach to Morphology. June 1988.
In: Trost, H.(ed.) 4. Österreichische Artificial-Intelligence-Tagung. Wiener Workshop - Wissenbasierte Sprachverarbeitung. Proceedings. Berlin etc.: Springer, 1988, 11-19.

Abstract: This paper presents an alternative approach to the use of the Finite State Automata for morphology in inflectional languages. The essential feature is the use of the morphological regularities of these languages to define a fine-grained word-class-specific subclassification. Morphological analysis and generation can be performed at the level of this classification by means of simple operations on n-ary trees. This approach has been implemented in the package MORPHIX which handles all inflectional phenomena of the German language. In spite of the complexity of the German inflection, the average time required to analyse a word is between 0.01 and 0.02 cpu-seconds.

Report No. 41
Harbusch, K.
Tree Adjoining Grammars mit Unifikation. June 1988.
In: Trost, H. (ed): 4 Österreichische Artificial-Intelligence-Tagung. Wiener Workshop - Wissenbasierte Sprachverarbeitung. Proceedings. Berlin etc.: Springer, 1988, 188-194.

Abstract: The formalism of Unification has become a main research topic in the field of natural language analysis. Unification is always bound to context free rules, although the question is, if there are other, larger units within the structure tree to which unification can be related. With the help of the Tree Adjoining Grammar formalism (TAG), these other, more complex units - namely subtrees of the structure tree - are specified and their combining capabilities defined.
In the following, I'll first give a brief description of the TAG and PATR formalism. The main emphasis of this paper though is placed upon defining the combination of these two formalisms, called Tree Adjoining Grammars with Unification. In the final section, certain advantages over both formalisms in their original state are discussed and open questions presented.

Report No. 42
Wahlster, W., M. Hecking, C. Kemke
SC: Ein intelligentes Hilfesystem für SINIX. August 1988.
To appear in: Gollan, W., Paul, W., Schmitt, A. (Eds.), Innovative Informationsinfrastrukturen, Informatik-Fachberichte Nr. 184, Berlin etc.: Springer, 1988.

Abstract: The SINIX Consultant (SC) is an intelligent help system for the SINIX operating system. SC represents a help system which combines a passive and an active mode in order to aid the user. In the passive mode, SC answers natural language questions about SINIX objects and commands in a user-oriented tutorial manner. In the active mode, it gives unsolicited advice to the user by detecting and correcting inefficient plans he is using. To fulfill these tasks, SC has to go beyond typical help system capabilities by providing natural language dialog facilities, a model of the user's knowledge, and a recognition of his actual plans and goals. The basis for performing the required problem solving and explanation tasks is a complex SINIX knowledge base describing commands and actions in the SINIX domain

Report No. 43
Wahlster, W.
Natural Language Systems: Some Research Trends. August 1988.
In: Schnelle, H., Bernsen, N.O. (eds.): Logic and Linguistics. Research Directions in Cognitive Science. European Perspectives, Vol. 2, Hove: Lawrence Erlbaum, 1989, 171-183.

Report No. 44
Kemke, C.
Representing Neural Network Models by Finite Automata. August 1988.
To appear in: Proceedings of the 1st European Conference on Neural Networks "Euro'88", Paris 1988.

Abstract: In this paper, networks of finite automata are suggested as a basic formalism for describing neural network models. Networks of finite automata are composed of automata with structured input and output sets - so-called Cartesian Automata. They represent, on the one hand, a clear, comprehensible formalism which is easy to handle mathematically. On the other hand, they are suitable as a basis for defining a wide class of neuron models. This fact is illustrated by comparing networks of finite automata with the general 'parallel distributed processing' model (given in Rumelhart, D.E. et al.: Parallel Distributed Processing, Vol. 1, MIT Press, Cambridge, MA 1986). Furthermore, a simple threshold neuron showing homeosthetic behavior, and an extended version of it providing learning and forgetting capabilities are described in terms of Cartesian Automata. Results out of theoretical investigations and computer simulations of this very general neuron model are outlined.

Report No. 45
Reddig, C.
"3D" in NLP: Determiners, Descriptions, and the Dialog Memory in the XTRA Project. August 1988.
In: Hoeppner, W. (ed.): Künstliche Intelligenz. GWAI-88. 12th German Workshop on Artificial Intelligence. Proceedings. Berlin, etc.: Springer, 1988, 159-168.

Abstract: A natural language processing system is only user-relevant if it can internally deal with the meaning of natural language utterances in a systematic way. For NL determiners, this paper contributes to this goal in two ways: It decomposes the determiner problem into its representational and interpretational aspects corresponding to the knowledge levels of the modular dialog system XTRA, and, by defining their complex meaning on these levels, it evaluates the determiners and determiner sequences possible in German and the descriptions formed by them.

World Watch

1.0 THEORETICAL ASPECTS

1245 Towards a general theory of information. II. Information and entropy.
T. Stonier
(Sch. of Sci. & Soc., Bradford Univ., UK).
ASLIB Proc. (UK), vol. 41, no. 2, p. 41-55 (Feb. 1989).

The author discusses the historical perspective on our perception of the world. He then considers the organization of matter, the fact that all organized structures contain information. He refers to the second law of thermodynamics and then examines the relationship between entropy and information. The significance of the Schrodinger equation is considered and information is modelled as an inverse exponential function of entropy. The communication engineers' view of information and entropy is discussed. Finally positive entropy, negative entropy and information magnitudes are discussed. In conclusion the author proposes a series of explorations that will hopefully lead to a genuine theory of information to aid the knowledge engineer. (16 refs.)

1251 How evaluation guides AI research.
P.R. Cohen, A.E. Howe.
Al Mag. (USA), vol. 9, no. 4, p. 35-43 (Winter 1988).

Evaluation should be a mechanism of progress both within and across Al research projects. For the individual, evaluation can show how and why methods and programs work, and so can show how research should proceed. For the community, evaluation expedites the understanding of available methods, and so their integration into further research. A five-stage model of Al research is presented and guidelines for evaluation that are appropriate for each stage are described. These guidelines, in the form of evaluation criteria and techniques, suggest how to perform evaluation. Recommendations that suggest how to encourage the evaluation of Al research are given. (27 refs.)

1260 Intelligent control for computer integrated manufacturing.
Y. Mohamed, L. Liu, M.A. Elbestawi, N.K. Sinha
(McMaster Univ., Hamilton, Ont., Canada.).
Proceedings of the 27th IEEE Conference on Decision and Control (IEEE Cat. No. 88CH2531-2), Austin, TX, USA, 7-9 Dec. 1988 (New York, NY, USA: IEEE 1988), p. 2314-21 vol. 3.

An intelligent machining controller is developed for cutting-force regulation of the peripheral milling process. This controller combines parameter adaptive control algorithms(s) with heuristic logic, which monitors the performance of the control algorithms. The development of knowledge and rule bases for intelligent control is discussed, and the resulting improvements in control performance are demonstrated using real-time experiments. (27 refs.)

1456 New approaches for heuristic search: a bilaterial linkage with artififcial intelligence.
F. Glover
(Graduate Sch. of Bus., Colorado Univ., Boulder, CO, USA), H.J. Greenberg. Eur. J. Oper. Res (Netherlands), vol. 39, no. 2, p. 119-30 (24 March 1989).

This survey considers emerging approaches of heuristic search for solutions to combinatorially complex problems. Artificial intelligence is a revived approach to probelm-solving that requires heuristic search intrinsically in knowledge-base operations, especially for logical and analogical reasoning mechanisms. Thus, one bilaterial linkage between operations reearch and artificial intelligence is their commomn interest in solving hard problems with heuristic search. But longstanding methods of directed tree search with classical problem heuristics, such as for the travelling salesman problem - a paradigm for combinatorially difficult problems - are not wholly satisfactory. Thus, new aproaches are needed, and it is at least stimulating that some of these are inspired by natural phenomena. (35 refs.)

1457 Am analysis of four uncertainty calculi.
S.J. Henkind, M.C. Harrison
(Dept. of Comput. Sc., New York Univ., NY,USA).
IEEE Trans. Syst. Man Cybern. (USA), vol. 18, no. 5,
p. 700-14 (Sept. - Oct. 1988). (received: 17 Apr 1989)

An important issue faced by contemporary artificial inteligence workers is how to deal with uncertain information. Four of the more prominent calculi-probability theory (especially the Bayesian apporach), the Dempster-Shafer theory, fuzzy set theory, and the MYCIN and the EMYCN calculi-are examined. Particular attention is paid to the underlying assumptions of these calculi and to their computational complexities. Each of the four calculi has a different perspective in uncertainty, and each manipulates uncertain information in a different way. Despite what some authors have claimed, there does not seem to be one calculus that is the best for all situations. Each of the calculi has its strong points; the main disadvantage seen in all of three calculi is that they compute aggregate numhers, but keep no record of divergence in opinions. (64 refs.)

1462 Nonmonotonic inheritance systems.
R.R. Yager
(Mach. Intelligence Inst. Iona Coll., New Rochelle, NY, USA). IEEE Trans. Syst. Man Cybern. (U.S.A.), vol. 18, no. 6, p. 1028-34 (Nov.-Dec. 1988). (received: 18 Apr 1989)

The use of frames and other hierarchical representational schemes has become failry pervasive in intelligent systems. Central to these representations is the inheritance of properties from one object to another by way of is-a-kind and is-an-element of links. Complexities and difficulties arise when the values inherited are typical (defeasible) and are transmitted through multiple links; in such cases, many of the issues inherent in nonmonotonic logics arise. Some procedures for making inferences in inheritance network that have both default (defeasible) and absolute knowledge (nondefeasible) are suggested. The basis for these suggestions is the use of possibility qualification for representing typical or default values. (14 refs.)

1706 Computational approaches to analogical reasonintg: a comparative analysis.
R.P. Hall
(Dept. of Inf. & Comput. Sci., California Univ., Irvine, CA, USA); Artif. Intell. (Netherlands) vol. 39, no. 1, p. 39-120 (May 1989).

Analogical reasoning has a long history in artificial inteligence research, primarily because of its promise for the acquisition and effective use of knowledge. Defined as a representational mapping from a known 'source' domain into a novel 'target' domain, analogy provides a basic mechanism for effectively connecting a reasoner's past and present experience. Using a four-component process model of analogical reasoning, this paper reviews sixteen computational studies of analogy. These studies are organized chronologically within broadly defined task domains of automated deduction, problem solving and planning, natural language comprehension, and machine learning. Drawing on these detailed reviews, a comparative analysis of diverse contributions to basic analogy processes identifies recurrent problems for studies of analogy and common approaches to their solution. The paper concludes by arguing that computational studies of analogy are in a state of adolescence: looking to more mature research areas in artificial intelligence for robust accounts of basic reasoning processes and drawing upon a long tradition of research in other disciplines. (114 refs.)

1707 The arts of the natural: Herbert Simon and artificial intelligence.
W.F. Baber
(San Diego State Univ., San Francisco, CA, USA).
Public Adm. Q. (USA), vol. 12, no. 3, p. 329-47 (Fall 1988).
(received: 02 May 1989)

Examines in some detail the central concerns of H. Simon's work since Administrative Behavior (1947). Although he has written a vast collection of material on a wide variety of subjects, his recurring theme during this period has been artificial intelligence or AI. But an empirical computer science is not Simon's ultimate goal. For Simon, a computer science is worth developing because of what it can reveal about the operation of the human mind. Both the computer and the human mind are symbol systems with symbol structures contained in physical patterns. Through time, both produce an evolving collection of expressions which serve as internal representations of the external environment to which the system must adapt. Both must use symbols to designate objects and relations. Finally, both must designate processes for interpretation and execution and these expressions must be stored in symbolic form. These common characteritstics lead Simon to conclude that any physical symbol system, be it brain or machine, has the necessary and sufficient means for intelligent action. (14 refs.)

2.0 SYSTEMS AND TECHNIQUES

1269 A method for evaluating candidate expert system applications.
J.R. Slagle, M.R. Wick.
AI Mag. (USA), vol. 9, no. 4, p. 44-53 (Winter 1988)

An evaluation method is developed for selecting expert system applications which are most likely to be successfully implemented. Both essential and desirable features of an expert system application are discussed. Essential features are used to ensure that the application does not require technology beyond the state of the art. Desirable features help point to the applications that stand the greatest chance for successful implementation. Advice on helpful directions for evaluating candidate expert system applications is also given. (16 refs.)

1294 Discovery of probabilistic rules for prediction.
K.C.C. Chan, A.K.C. Wong
(Dept. of Syst. Design Eng., Waterloo Univ., Ont., Canada), D.K.Y. Chiu.
Proceedings. The Fifth Conference on Artificial Intelligence Applications (IEEE Cat. No. 89CH2712-8), Miami, FL, USA, 6-10 March 1989 (Washington, DC, USA: IEEE Comput. Soc. Press 1989), p. 223-9.

An inductive learning algorithm is presented for analysing the inherent patterns in a sequence and for predicting future objects based on these patterns. This algorithm is divided into three phases: detection of underlying patterns in a sequence of objects; construction of rules, based on the detected patterns, that describe the generation process of the sequence; and use of these rules to predict the characteristics of the future objects. The learning algorithm has been implemented in a program known as the OBSERVER, and it has been tested with both simulated and real-life data. The experimental results show that the OBSERVER is capable of discovering hidden patterns and explaining the behavior of certain sequence-generating processes that a user is not immediately aware of or fully understood. For this reason, the OBSERVER can be used to solve complex real-world problems where predictions have to be made in the presence of uncertainty. (20 refs.)

1302 RTEX: an industrial real-time expert system shell.
A.R. De Feyter
(A.I. Systems NV/SA, Brussels, Belgium).
Proceedings of AIDA-88. Fourth Annual Conference on Artificial Intelligence and Ada, Fairfax, VA, USA, 15-16 Nov. 1988 (Fairfax, VA, USA: George Mason Univ. 1988), p. 6/1-22.

RTEX is an industry oriented software development system for creating embedded real-time expert systems in a possibly distributed hardware environment. Written in the Ada language it integrates the most advanced concepts used in real-time distributed software systems with many features of the well established AL tools. Object oriented programming, parallelism, real-time inferencing, symbolic matching and signal understanding are key capabilities of the system. Target application domains are process control, telecommunication switching and network management, manufacturing control, aircraft and naval navigation, missile and satellite guidance, industrial robots and simulation. (17 refs.)

1303 ERS: an expert system shelll designed and implemented in Ada.
S.H. Hirshfield
(Hamilton Coll., Clinton, NY, USA).
Proceedings of AIDA-88. Fourth Annual Conference on Artificial Intelligence and Ada, Fairfax, VA, USA, 15-16 Nov. 1988 (Fairfax, VA, USA: George Mason Univ. 1988), p. 10/1-34

Describes a project originally limited to porting an existing expert system shell, the embedded rule-based system (ERS), to Ada. Much to the authors' surprise the project evolved into a major redesign of ERS which exploits Ada's facilities for data abstraction and object-oriented development. The resulting Ada implementation has all of the functionality of earlier versions of ERS (with hooks for many additional features), maintains upward compatibility with existing rule bases, is significantly more efficient than previous versions, and is of higher overall quality by any software engineering standards. Most important, the project demonstrates, convincingly, Ada's suitability and utility for developing knowledge-based systems and embedded AI applications in general. (14 refs.)

1317 Verification and validation of rulebased systems for Hubble Space Telescope ground support.
S. Vick, K. Lindenmayer
(Space Telescope Sci. Ins., Baltimore, MD, USA).
Telemat, Inform. (UK), vol. 5, no. 3 , p. 313-23 (1988).
(1988 Goddard Conference on Space Applications of Artificial Intelligence, Greenbelt, MD, USA, 24 May 1988).

As rulebased stystems become more widely used in operational environments, one must begin to focus on the problems and concerns of maintaining expert systems. In the conventional software model, the verification and validation of a system have two separate and distinct meanings. To validate a system means to demonstrate that the system does what is advertised. The verification process refers to investigating the actual code to identify inconsistencies and redundancies within the logic path. In current literature regarding maintaing rulebased systems, little distinction is made betwee these two terms. In fact, often the two terms are used interchangeably. The authors discuss verification and validation of rulebased systems as separate but equally important aspects of the maintenance phase. They also describe some of the tools and methods that developed at the space Telescope Science Institute to aid in the maintenance of the rulebased systems. (13 refs.)

1502 Experiments with experts developing simple expert systems.
M. King
(Loughborough Univ. of Technol., UK)
Omega (UK), vol. 17, no. 2, p. 123-34 (1989)

It has been suggested that current expert systems technology enables experts to develop their own small systems. This hypothesis is investigated by observing four experts in different fields as they develop their own prototypoe expert systems. The four experts had little previous experience of expert systems and tackled the projects as part of a postgraduate course using a readily available expert systems shell. The results of the four cases, which are described in detail, vary considerably. One resulted in a successful prototype which is being further developed into an operational system. Another highlighted the need for technical assistance at several stages of the development process. It is concluded that there is considerable potential for experts to develop their own simple systems but some will need support in using shells and structuring their knowledge. (15 refs.)

1508 Data processing expert systems: the DAPES Club.
J. Griffin
(Metropolitan Police Office, London, UK, J. Heslop.
KBS in Government. Proceedings of the Conference. Gatwick, UK, 4-5 Nov. 1987 (Pinner, UK: Online Publications 1987), p. 37-47.

The ALVEY-sponsored DAPES Club has been in action since early 1986. Its members are a collection of major organisations with DP deparments, two commercial collaborators and two academic establishments. This paper attempts to pass on the experience the Club has gained in building DP expert systems for use by DP people. In line with the Club's project structure, this falls naturally into two main phases. Phase I gave particular experience in the areas of prototyping, feasibility and and incremental development. Phase II has given rise to development of an operational system, and the use of an intermediate representation. The author describes the approaches used for knowledge acquisition and elicitation.

1511 Computer-based critics.
G. Fischer, T. Mastaglio
(Dept. of Comput., Sc., Colorado Univ., Boulder, Co., USA). Proceedings of the Twenty-Second Annual Hawaii International Conference on System Sciences Vol. IIII: Decision Support and Knowledge Based Systems Track (Cat. No. 89THO244-4), Kailua-Kona, HI, USA, 3-6 Jan. 1989 (Washington, DC, USA: IEEE Comput,. Soc. Press 1989), p. 427-36, vol. 3.

The authors describe computer-based critics and articulate some of the general principles learned from their system-building experience. They propose a general framework for critics, present specific requirements, and describe two prototypical critic systems: LISP-CRITIC, which criticizes Lisp programs, and CRACK, a system that assists the user in designing a kitchen. The authors illustrate the generalized main componnents of the critic systems and discuss their evaluation. It is concluded that computer-based critics incorporate many powerful ideas from human-computer communications and artificial intelligence into a system that makes use of the best aspects of human and computational cognition. They have the potential to provide a symbiotic relationship between a user and a knowledge-based system. The results should be applicalbe to the entire class of cooperative problemsolving systems. (28 refs.)

1516 Knowledge-based approach to the formulation of ultrasonic nondestructive testing procedures.
A. McNab, H.S. Young
(Dept. of Electron. & Electr. Eng., Strathclyde Univ., Glasgow, UK)
IEE Proc. A. Phys. Sci. Meas. Instrum, Manage. Educ. (UK), vol. 136, no. 3, p. 134-40 (May 1989)

In nondestructive testing there is considerable potential for the application of knowledge-based systems. This ranges from the selection of suitable testing techniques, through the formulation of test procedures, to the analysis and interpretation of test results. In this particular case, an expert system shelll was used to develop a system which assists in the derivation of procedures for the ultrasonic inspection of welds in ferritic steel. The components of the system are described and examples given of its use and subsequent outputs. The advantages of such an approach are improvements in the speeed and accuracy of procedure writing, especially for less experienced personnel, the potential as a training aid and the ability to document the system consultation. (5 refs.)

1518 Nine problems in artificial intelligence.
R.C. Schank, C.C. Owens
(Yale Univ., New Haven, CT, USA).
Data and Knowledge (DS-2). Proceedings of the Second IFIP 2.6 Working Conference on Database Semantics, Albufeira, Portugal, 3-7, Nov. 1986 (Amsterdam, Netherlands: North-Holland 1988), p. 321-40

Problems in AI have typically been characterized at the wrong level of abstraction and there are certain fundamental problems that transcend specific tasks like chess playing or langage translation. It is against these underlying problems and not against specific tasks that one should measure the progress of research in the field. The authors outline a few of the problems they consider to be essential and fundamental. They are knowledge representation, decoding, inference, control of combinatorial explosion, indexing, prediction and recovery, dynamic modification, generalization and creativity. (12 refs.)

1548 Expert system for patient realignment in MRI (magnetic resonance imaging).
H.S. Young
(Dept. of Electr. & Electron., Eng., Strathclyde Univ., Glawsgow, UK), B. Dalton, N. Saeed, S. Marshall, T.S. Durrani.
IEE Colloqauium on "The Application of Artificial Intelligence Techniques to Signal Processing (Digest No. 42), London, UK, 15 March 1989, London, UK:IEE 1989), p.8/1-3

Describes an approach for carrying out automatic patient realignment using a completely separate knowledge based system which calls on a series of low level image/signal analysis tools to extract low level features from the images. These features are compared using the knowledge based expert system to measure and then correct the positional inaccuracies. Higher level features such as the labryinth and individual sulci can then be extracted and used to fine tune the realignment. While the feature extraction and image analysis operations are implemented in a standard numberical programming language, the symbolic reasoning is carried out using an expert system shell. The authors discuss some of the reasons for adopting this type of approach and describe how breaking the problem up as an intelligent overseer driving a series of low level image processing algorithms, has led and guided the work.

1736 How to identify expert systems opportunities.
B.O. Szuprowicz.
Can. Datasyst. (Canada), vol. 21, no. 3, p. 52-3 (March 1989)

Much has been said recently about the use of expert systems to enhance productivity and profitability of corporate activities, but there is little information about the most practical approaches to identify strategic expert systems opportunities. Although these procedures will undoubtedly differ from one corporate culture to another there are nevertheless certain general principles that are quite universal and can assist company executives in eliminating much of the risk inherent in the introduction of an untried and unknown new technology.

1740 Value-driven expert systems for decision support.
R.L. Keeney
(Dept. of Syst. Sci., Univ. of Southern California, Los Angeles, CA, USA).
Decis. Support Syst. (Netherlands), vol. 4, no. 4, p. 405-12 (Dec. 1988)

Values are an inherent part of all decision processes. Hence, values are at least implicity included in all expert systems intended for decision support. This paper outlines the concepts and methodology, which are based on the principles and procedures of decision analysis, to address explicily the values in an expert system logically and consistently. Implementation of the concepts and methodology involves the elicitation of values using the same general approach as that used by knowledge engineers to explicate expert knowledge. (34 refs.)

1743 Expert systems and behavioral decision research.
D. von Winterfeldt
(Inst. of Safety & Syst. Manage., Univ. of Southern California, Los Angeles, CAL, USA).
Decis. Support Syst. (Netherlands), vol. 4, no. 4, p. 461-71 (Dec. 1988)

This paper reviews a part of the literature on behavioral decision research (policy capturing, psychophysics of numerical judgments and cognitive illusions) and examines implication for knowledge elicitation in expert systems. The literature on policy capturing demonstrates that simple and compact numerical models of expert knowledge can be built, but that experts are poor in verbalizing the knowledge expressed in them. The psychophysical literature indicates that numerical encoding of expert knowledge may be difficult and biased, but that it has definitive advantages over qualitative elicitation schemes: Inumerical encoding forces hard thought, encourages precision, and allows to access a substantial computational apparatus. The literature on cognitive illusions suggests that the expert knowledge one elicits may be an illusion. The review concludes by recommending to use numerical judgments and explicit models by experts where possible, and to decompose the elicitation task in order to avoid cognitive illusions. (48 refs.)

1755 Testing algorithms and programs of multivariate statistical procedures-necessary assumption of building expert systems.
E.-M. Tiit
(Tartu State Univ., Estonia SSR, USSR).
Int. J. Pattern Recognit. Artif. Intell. (Singapore), vol. 3, no.1, p. 121-33) (March 1989)

Sometimes, programs for multivariate statistical procedures are included in expert systems. The requirements of accuracy, exactness and reliability for such programs are very high. A method for testing algorithms and programs of multivariate statistical procedures, the 'exact samples method', is introduced. The programs of simple linear regression analysis from four of the most popular standard packages are tested and compared with the help of the methods. (9 refs.)

1757 Architecture of a coupled expert system for optimum design of plate girder bridges.
H. Adeli, K.Y. Mak
(Dept. of Civil Eng., Ohio State Univ., Columbus, OH, USA).
Eng. Appl. Artif. Intell. (UK), vol. 1, no. 4, p. 277-85 (Dec. 1988).

A prototyope knowledge-based expert system has been developed for optimum design of steel plate girders used in highway bridges. This expert system, called PG-BRIDGE1, is a coupled system in which AI based symbolic processing is combined with the traditional numberical processing. Plate girders can be unstiffened or stiffened with single- or double-sided transverse stiffeners. They can be homogeneous or hybrid, made of high-strength flange plates and low-strength web plate. A mathematical optimization algorithm has been developed for minimum weight design of plate girders using the generalized geometric programming technique. The basis of design is the American Association of State Highway and Transportation Officials (AASHTO) specifications. The plate girders are subjected to the live (moving) loads of the AASHTO specifications. The knowledge base and symbolic processing has been developed using the Expert System Development Environment (ESDE). Numerical processing for structural analysis, optimization algorithm and graphics interface have been developed in FORTRAN 77 (20 refs.)

1776 OPERA - an expert operations analyst for a distributed computer system.
R.M. Adler
(Mitre Corp., Bedford, MA, USA). A. Heard, R.B. Hosken. Proceedings of the Annual A1 Systems in Government Conference (IEEE Cat. No. 89CH2715-1), Washington, DC, USA, 27-31 March 1989 (Washington, DC,USA: IEEE Comput. Soc. Press 1989), p. 179-85

OPERA (operations analyst) is a suite of expert systems being developed to improve operations support to a complex distributed computer network in the Space Shuttle Launch Processing System. The two expert systems in the initial OPERA prototype assist users in isolating and correcting network problems. One expert system monitors network error messages, which interprets to hypothesize faults and to suggest troubleshooting and problem recovery procedures. The second system searches a knowledge base containing symgbolic digests of error tracking databases, retrieving relevant precedents to network anomalies (detected by the first system). Expert systems are integrated into O)PERA by embedding them in blackboard structures. A controller blackboard routes communications among esxspert systems and coordinates their activites. The architecture supports shard access to external interface as well as domain knowledge bases, and is highly extensible. (13 refs.)

3.0 APPLICATIONS

1326 Self-help legal software and the unauthorized practice of law.
R.D. Vincenti.
Comput/Law J. (USA), vol. 8, no. 2, p. 185-226 (Spring 1988). (received: 23 Feb. 1989)

Describes some of the self-help legal software that is available to consumers, discusses the unauthorized practice of law, and determines whether the publishing and sale of such software constitutes the unauthorized practice of law. The first section discusses how the software, through deductive reasoning, performs an analysis of the particular user's problem. The advantages and disadvantages are explored and the software is compared to other types of self-help legal materials available to consumers. The second section explains relevant aspects of the unauthorized practice of law. An analysis of relevant case law is included and the policy reasons for preventing the unauthorized practice of law are presented. Finally, the author analyzes self-help legal software in light of the statutes and cases discussing the unauthorized practice of law. Because the software provides what appears to be an analysis of the user's problem, unlike traditional sources of self-help information, and may be of quesitonable accuracy in all situations, self-help legal software should be banned as the unauthorized practice of law. (135 refs.)

1328 A Role of the expert system in project management.
P. Robinson
(Westpac Banking Corp., London, UK)
Int. J. Proj. Manage. (UK), vol. 7, no. 1, p. 25-8 (Feb. 1989)

According to a recent survey conducted by PA Management Consultants, more than 50% of UK companies are either using or developing computer systems which mimic human thought processs. Of those companies surveyed, nine out of ten will be using 'expert systems' within the next three years. From a financial perspective the Ovum research group estimates that the UK expert system market will expand from a magnitude of ?40M in 1986 to ?400M by 1992 and that the European market will grow to over ?3000M to 1990. The paper examines the role of expert systems in project management, beginning with a discussion of the components of a typical expert system and identification of the fundamental keys to successful project management. Knowledge acquisition poses problems; two basic approaches are given. Selection of a good expert system shell is detailed. (9 refs.)

1344 A comparison of analytic and knowledge-based approaches to closed-shop scheduling.
P. Alpar, K.N. Srikanth
(Dept. of Inf. & Decision Sc., Coll. of Bus. Adm., Illinois Univ., Chicago, IL, USA)
Ann Oper. Res. (Switzerland) vol. 17, no. 1-4, p. 347-62 (Jan. 1989)

This paper compares three different approaches to scheduling in a closed-shop environment, making the case for a knowledge-based approach. A manufacturing example from the food industry is used as a vehicle for the presentation. The first approach attempts to find an optimal solution using a mixed integer linear programming formulation, but the size of the problem renders this approach impractical. The second approach uses a spreadsheet program to obtain feasible solutions, but imbedded assumptions in the heuristics used allow it to be used only for simple demand patterns. The third approach employs expert systems technology. It includes seeveral heuristics and takes all constraints into consideration. The solution obtained may not be optimal, but computational tests suggest that it is far superior to both spreadsheet and manual approaches. (12 refs.)

1353 HEMAVID: a flexible computer-based interactive video resource for hematology.
H.C. Chueh, G.O. Barnett, W.C. Hayes, W.S. Beck
(Lab. of Comput. Sci., Massachussetts Gen. Hospital, Boston, MA, USA).
Proceedings. The Twelfth Annual Symposium on Computer Applications in Medical Care (IEEE Cat. No. 888CH2616-1), Washington, DC, USA, 6-9 Nov. 1988 (Washington, DC, USA: IEEE Comput. Soc. Press 1988), p. 421-5.

The design and development of HEMAVID, a computer-based interactive video (CBIV resource for hematology, is described. HEMAVID is a PC-based program that builds and maintains a versatile object-oriented database that indexes a collections of hematology slides residing on videodisc. Access to the database is flexible, allowing its use as: (1) an interactive visual reference; (2) a nucleus for the development and presentation of computer-aided instruction (CAI), and (3) image content in patient simulations. Each image in HEMAVID is associated with expert description a prerequisite for the program's primary target population of medical students. Research fellows and physicians should also find it an effective resource for hematology review. To that end, the program incorporates an intuitive user interface; controlling the program requires only a mouse as a pointing device. HEMAVID's integration of expert knowledge and video-imaging technology provides a unique alternative to traditional methods of teaching and reviewing hematological morphology. (18 refs.)

1361 PEPTY: a knowledge-based program for assisting medical reasoning in peptic diseases.
M. Torchio, G. Molino, A. Cavanna, L. Appendini, A.M. Fornara
(Departimento di Biomedicina, Clinnica Medica E, Torino Univ., Italy).
Comput. Methods Programs Biomed. (Netherlands, vol. 28, no. 4. p. 249-56 (April 1989).

PEPTY is a program developed with the aim of providing a diagnostic and therepeutic assistance in managing peptic diseases. Its theoretical basis is an accurate analysis of current concepts in peptic ddisease diagnosis and treatment. This was done by reviewing recent liaterature and consulting skilled gastroenterologists. The decision tree includes three sections dealing with diagnostic, therapeutic and monitoring problems. The diagnostic section starts by evaluating clinical data from patient history and physical examination; the diagnostic hypotheses given at this level are refined and eventually confirmed by further information in the following section. Here the decision tree becomes modular in that a proper therapeutic and monitoring pathway is defined for four disease classes: gastroduodental peptic ulcer and duodenitis, gastro-oesophageal reflux, erosive gastritis, and chronic antral gastritis. In the therapeutic section a cost-benefit analysis of possible therpeutic choices is always performed, but the final decision is made by the user. Complications, side effects and treatment efficacy are also considered and the program finally suggests the appropriate maintenance treatment. Patient data display, storage and retrieval, and explanation faciliites are supplied. The system can provide a 'second opinion' in the medical practice and may be a useful learning tool for medical students. (16 refs.)

1367 Evaluating the utility of available differential diagnosis systems.
J.R. Hammersley, K. Cooney
(Dept. of Internal Med., Michigan Univ. Med. Center, Ann Arbor, MI, USA).
Proceedings. The Twelfth Annual Symposium on Computer Applications in Medical Care (IEEE Cat. No. 88CH2616-1), Washington, DC, USA, 6-9 Nov. 1988 (Washington, DC, USA: IEEE Comput. Soc. Press 1988), p. 229-31

Results of an evaluation testing the accuracy and performance of two differential diagnosis systems. Meditel and DXplain, obtained by utiliszing 103 consecutive clinical cases are presenteed. Each case was run in parallel on both systems utilizing available history, physical and lab findings at admission and comparing the resultant differential list to the final hospital discharge diagnoses. It was found that these systems use fundamentally different approaches to the problem of differential diagnosis, which produces significant differences in their speed and in the usability of their differential lists. Meditel produces generally more accurate and complete differential diagnosis lists, faster than DXplain, and has the advantage of local use on a PC. DXplain is more user-friendly in its interaction, and has a better explanation facility, but appears less able to separate the multiple diagnoses that patients often exhibit. (2 refs.)

1368 Determination of testing efficacy in carcinoma of the lung using a neural network model.
D.L. Hudson
(California Univ., San Francisco, CA, USA). M.E. Cohen, M.F. Anderson.
Proceeedings. The Twelfth Annual Synposium on Computer Applications in Medical Care (IEEE Cat. No. 88CH2616-1). Washington, DC, USA, 6-9 Nov. 1988) (Washington, DC, USA: IEEE Comput. Soc. Press 1988), p. 251-5.

A Neural network approach to expert system generation is presented. The goal of the system is to aid in the determination of effective testing straregies for treatement of carcinoma of the lung. A retrospective study was undertaken to determine important factors in treatment decisions. These factors were then used in a supervised learning algorithm to produce a neural network model which subsequently can be used as an expert system. The neural network supervised learning approach to expert system generation allows a more automated production of the knowledge base. Preliminary results indicate that this type of analysis can be used effetively in determining efficacy of testing strategies, both for diagnosis and treatment of carcinoma of the lung, as well as other diseases. (29 refs.)

1370 Braindex - an expert system for the diagnosis of brain death.
G. Pfuratscheller
(Tech. Univ. Graz, Austria), G. Schwarz, H. Moik, V. Haase. Biomed. Tech. (West Germany), vol. 34, no. 1-02, p. 3-8 (Jan.-Feb. 1989). In German.

An IBM PC-based system was developed as a decision-supporting system for the diagnosis of brain death. Braindex has been realized with the expert system Tool PC-Plus, and includes a knowledge base containing 300 rules organized in eight frames. Both forward chaining and backward chaining are used in the inference engine. Braindex also includes a text archive and a lexicon. The data of 42 brain death patients are stored in the data bank (300 parameter per patient). The system is presently in the first stage of clinical application. (8 refs.)

1379 Expert systems and weather forecasting.
B.J. Conway
(Meteorol. Office, Bracknell, UK).
Meteorol. Mag. (UK), vol. 118, no. 1399, p. 23-30 (Feb. 1989).

The paper explains what expert systems are and how they differ from conventional computer programs. It looks at the special demands weather forecasting makes on such systems and outlines the work which has started in the Meteorologicial Office to apply expert systems to short-period weather forecasting. Pilot projects on nowcasting precipitation and forecasting thunderstorms are discussed. (22 refs.)

1396 Selecty-HVAC: knowledge-based system as an advisor to configure HVAC sysstems.
P. Fazio, R. Zmeureounu
(Centre for Building Studies, Concordia Univ., Montreal, Que., Canada)., A. Kowalski.
Comput. Aided Des. (UK), vol., 21, no. 2, p. 79-86 (March 1989)

A knowledge-based system is presented that has been developed to be used as an advisor in the preliminary design stage of HVAC systems. It enables designers to configure and size HVAC equipment for different climatic conditions, in terms of building type, indoor requirements, outdoor and indoor air pollution. A new approach has been used in building the system, using a modified knowledge engineering methodology and a menu-driven front-end to a full frame-based system. (27 refs.)

1397 Expert inspector of surface defects.
H.A. ElMaraghy, D.J. Bullis
(Center for Flexible Manuf. Res. & Dev., McMaster Univ., Hamilton, Ont., Canada).
Comput., Ind. (Netherlands), vol. 11, no. 4, p. 321-31 (Feb. 1989).

An automated system for inspecting automotive engine valves has been designed and implemented. The design was in two parts: a real-time, vision-based surface inspector, and an off-line expert defect classification and diagnosis system. The real-time inspection cell consists of: a solid-state camera, an optical shaft encoder, and an IBM/AT computer. The inspection cell provides real-time accept/reject decision making, and allows operator interactive inputs. The expert defect classification and diagnosis system was implemented on a MICRO VAX II in FORTRAN-77. It used information from the vision inspection cell and a knowledge base extracted from the valves manufacturer in inspection and production experts to provide classification and off-line tracing of defects to their sources. The hardware and software structure of the visual inspection system are described. In addition, the knowleege base and expert rules ued in defect classification and diagnosis are described, and outputs for typical valve defects are included. (6 refs.)

1415 Application of the expert system to elevator group supervisory control.
S. Tsuji, M. Amano
(Mitsubishi Electr. Corp., Aichi, Japan), S. Hikita.
Proceedings. The Fifth Conference on Artificial Intelligence Applications (IEEE Cat. No. 89CH2712-8) , Miami, FL, USA, 6-10 March 1989 (Washington, DC, USA: IEEE Comput. Soc. Press 1989), p. 287-94.

A novel elevator group-supervisory control method to which an expert system and fuzzy rules are applied, is suggested. This method determines the optimum car to answer a hall call by using knowledge bases, production rules and fuzzy rules, extracted from the knowledge of experts, in elevator group-supervisory control. It has been verified through simulation that the method can bring about considerable improvement, reducing the average waiting time by 15-20%, and long waits (not less than 60s) 30-50% in comparison with the conventional method, in which a fixed evaluation function is used. Knowledge-based systems suitable to the online and real-time control for the implementation of this method on 32-b microcomputers are suggested. The result of the simulation has demonstrated that the suggested knowledge-based systems satisfy the required response time of 100-150 ms in the group-supervisory control system with an immediate prediction function. (5 refs.)

"Based on the information provided in the abstracts, the references provided have been selected by the secretariat of the NRC Associate Committee on Artificial Intelligence as a representative sample of interest and value to Canadian industry. Abstracts provided are reprinted from "Key Abstracts in Artificial Intelligence" with premmission from INSPEC. INSPEC is widely recognised as the leading English-language database covering the published information in the field of physics, electronics and computing. Information contained in the INSPEC services is collected on an international basis from over 4,0000 Journals and 1,000 Conference Proceedings. INSPEC is a division of the Institution of Electrical Engineers, Station House, Nightingale Road, Hitchin, Herts, United Kingdom. All INSPEC's products and services are available in North America from the INSPEC Dept. IEEE Service Centre, 445 Hoes Lane, P.O. Box 1311, Piscataway, N.J. 08855-1331, U.S.A."

Persons wishing to obtain copies of references cited should contact their nearest technical library or the Canada Institute for Scientific and Technical Information, National Research Council, Building M-55, Montreal Road, Ottawa, Ontario, CDA, K2A 0S2, (tel.: (613) 993-1585, Telex 053-0315.) For on-line ordering, CAN/OLE users may use the CAN/DOC command. Envoy users type "COMPOSE CISTI."

1552 Welfare benefits adviser: a local government expert system application.
J. Sutcliff.
Comput,. Law. Secur. Rep. (UK), vol. 4, no. 6, p., 22-3 (March-April 1989)

Walsall MBC, in conjunction with ICL, are developing an expert system to give advice on UK welfare benefits. The system will advise potential claimants on their entitlement, ini6tially to Income Support, Family Credit, Mobility Allowance, Attendance Allowance and Invalid Care Allowance, with other benefits to be added later on. Although this version covers only 5 of the approximately 70 benefits, in fact these 5 benefits account for about 85% of claims to the DSS. The system is on field trial in six of the Council's neighbourhood offices. The author summarises developments so far, by discussing the concept of the neighbnourhood office, computerised benefit advice and the future of the system.

1556 The COCOM advisory system.
M.J.R. Keen
KBS in Government.
Proceedings of the Conference, Gatwick, UK, 4-5, Nov. 1987 (Pinner, UK: Online Publicaitons 1987), p. 49-57.

The Security Expert Control regulations-known as the COCOM regulations-govern the expert of goods to Eastern Block countries. These regulations are extremely complex and their administration by the Department of Trade and Industry requires a team of skilled technical officers. As the first stage in introducing the use of expert systems into this area, a demonstration expert system has been built to show how such techniques could assist with the administration of the regulations. This demonstration implements Item ILI565, the largest and most complex item of the COCOM regulations, which deals with the export of computers and associated equipment and systems.

1560 An interactive graphical interface for an expert airload-planning system.
J.S. Wong
(SRI Int., Menlo Park, CA, USA).
Proceedings of the Twenty-Second Annual Hawaii International Confernce on System Sciences, Vol. IV: Emeging Technologies and Applications Track (IEEE Cat. No. 89TH0245-1), Kailua-Kona, HI, USA, 3-6 Jan. 1989 (Washington, DC, USA: IEEE Comput. Soc. Press 1989), p. 22-7 vol. 4

A graphics system is designed and implemented as an adjunct to an expert system for planning aircraft loading. This system is fully integrated with the expert system and is used to illustrate its planned airloads and to modify these loads interactively. Design considerations and key features of the graphics system are discussed. The graphics system has been deployed successfully and now widely accepted as a useful tool for airload planning and execution. Aspects of the system that could be improved are indicated. (3 refs.)

1568 ESKORT: an expert system for auditing VAT accounts.
H.K. Jacobsen
(Comput. Res. Int. A/S, Copenhagen, Denmark).
KBS in Government. Proceedings of the Conference, Gatwick, UK, 4-5 Nov. 1987 (Pinner, UK: Online Publications 1987), p. 131-46.

CRI A/S has developed an expert system for auditing VAT accounts for the Danish Customs Authorities. A key issue is that the system integrates a number of expert systems with traditional data processing systems working on the same set of data. The result is not just another expert system prototype but a system which is able to support the complex task of auditing VAT accounts. A task which requires handling a great amount of data as well as expert abilities to diagnose errors and to plan reasonable actions.

1610 Experiences with an automatic transfer function algorithm.
D.P. Reilly, K. Dooley
(Automatic Forecasting Syst. Inc., Hatboro, PA, USA).
Computer Science and Statistics: proceedings of the 19th Symposium on the Interface, Philadelphia, PA, USA, 8-11 March 1987 (Alexandria, VA, USA: American Stat. Assoc. 1987), p. 128-35

Transfer function models are extremely useful for representing time series relationships and thus for forecasting time series data.

The authors discuss an expert system for the development of these types of models. The algorithm is evaluated by applying it to a number of time series that have already been used as examples for transfer function modelling. (9 refs.)

1616 Cell placement expert system (K/cp) for VLSI design.
R. Hamazakiu, N. Shiraki
(Div. of Res. & Dev., OKI Electr. Ind. Co., Ltd., Tokyo, Japan). Oki Tech. Rev. (Japan), vol. 54, no. 130, p. 11-15 (May 1988). (received: 04 Apr 1989)

A CAD system for VLSI design has been developed by using artificial intelligence (AI) to improve the productivity of application specific integrated circuit (ASIC) design. As an interm result, a rule-based cell placement expert system was developed and was put into practical use together with a conventional CAD system. This practical usage proved that the placement capability of the expert system would surpass the manual placement capability in the case of larger circuits. It was also demonstrated that processing that originally was difficult to achieve by the conventional method could be realized easily. Thus, a clear prospect was obtained that highly integrated and high-performance LSIs could be designed with the CAD system using AI. This paper outlines the expert system and explains the method of realizaiton and the evaluation results. (6 refs.)

1623 Sophisticated firmware creates intelligent calibrators.
M. Carter
(John Fluke Manuf. Co., Inc., Everett, WA, USA).
Eval. Eng. (USA), vol. 28, no. 1, p. 80-2 (Jan. 1989).

The increasing complexity and accuracy of test equipment poses tough challenges for calibration instruments. Calibrators need to be more accurate than ever to cover the breadth of today's workload. To keep labor and training costs down, calibrators also need to be easy to operate. And, to help companies stay competitive, ownership costs must be kept low. The goal of high accuracy appears to conflict with the other two. To allow all three deign goals to be met, this article presents a solution: building intelligence into the calibrator via internal firmware, which allows the calibrator quickly and easily to check its drift between formal calibrations, increasing user confidence in its ability to remain in tolerance over long periods.

1657 Cockpit crew scheduling and supporting system using AI techniques.
K. Onodera
(2nd EDP Local Gov. & Public Utility Syst. Div., NEC Corp., Tokyo, Japan).
NEC Res. Dev. (Japan), no. 92, p. 151-6 (Jan. 1989).

When scheduling cockpit crews, various constraints must be observed. In addition, changes in the system environment such as the introduction of new aircraft or the alterations in labor agreements must be dealt with. OR (operations research) techniques have been frequently used for the combination problem such as scheduling or planning, but it is difficult to use this technique for cockpit crew scheduling because of the difficulties involved in formulation and sofware maintenance. To deal with this problem, the author has developed a scheduling expert system in cooperation with Japan Airlines Co., Ltd. by using AI techniques and incorporating the heuristics of the specialists who prepare the scheduling.

1660 A simulation study of expert control system for flotation.
S.L. Jamsa, J.A. Herbst
(Dept. of Metall, Utah Univ., Salt Lake City, UT, USA).
Software for Computer Control 1988 (SOCOCOCO '88): Fifth IFAC/IFIP Symposium, Johannesburg, South Africa, 26-28 April 1988 (Oxford, UK: Pergamon 1989),p. 67-76

Computer aided control system design (CACSD) has become very common during the last ten years. Even the use of these software packages for the design of the contol strategies which use the conventional single input single output controllers has shown to be a powerful tool for a control engineer. In the paper a simulation study of control strategies for the flotation process is presented. The derivation of a dynamic model to simulate the flotation process is described. The interfacing of this simulator with the ON SPEC control software package is discussed. Differences between the classical PID-control strategies and the expert strategy designed using this software for the flotation process are discussed. (14 refs.)

1670 Successful expert systems for Space Shuttle payload integration.
K. Morris
(Rockwell Int., Downy, CA, USA).
Fourth Conference on Artificial Intelligence for Space Applications (NASA Conf. Publ. 3013), Huntsville, AL, USA, 15-16 Nov. 1988 (Washington, DC, USA: NASA 1988), p. 245-52.

Expert systems have been successfully applied to solve recurring NASA Space Shuttle orbiter payload integration problems. Recurrence of these payload integration problems is the result of each Space Shuttle mission being unique. The NASA Space Shuttle orbiter was designed to be extremely flexible in its ability to handle many types and combinations of satellites and experiements. The first successful expert system to be applied to these problems was the Orbiter Payload Bay Cabling Expert System (EXCABL). The operational version of EXCABL successfully solved the payload electrical support services cabling layout problem. A second expert system, Expert Drawing Matching System (EXMATCH), was developed to generate a list of the reusable installation drawings available for each EXCABL solution. EXMATCH was delivered for operational usage in 1987. As a result of these initial successes, the need for a third expert system was defined and awaiting development. The new expert system, called Technical Order Listing Expert System (EXTOL), will generate a list of all the applicable reusable installation drawings available to support the total payload bay mission provisioning and installation effort. The paper describes these expert systems, the individual problems that they were designed to solve, their individual solutions and the degree of success they have achieved. These expert systems' successes instantiate the applicability of this technology to the solution of real-world Space Shuttle payload integration problems. (3 refs.)

1679 Expert system in village water supply design.
P. Poyet., M. Detay.
Report 936, Inst. Nat. Recherche Inf. Autom., Le Chesnay, France (Dec. 1988), 37 pp. in French.

The system described is a PC designed expert system for village water supply programmes. The aim was to develop efficient software, running on hardware available in the developing countries, with a high level of expertise in the field of village water supplies programmes. The hardware constraints lead to a specialized and sophisticated software architecture. The underlying mechanisms of HYDROLAB do not rely on fully generic schemes, but rather on solutions to application dependant problems. As far as the user is concerned, the system is acting as a diagnostic expert system, asking questions, analysing the user answers, and building plans in order to schedule the appropriate set of actions and to explore constrained parts of a large search tree. The system is able to make a diagnosis about the user situation, to give advice in order to increase the average success, and to evaluate the hydrodynamic characteristics of the future well according to experience based on the statistical processing of 1080 drillings in Africa (35 refs.)

"Based on the information provided in the abstracts, the references provided have been selected by the secretariat of the NRC Associate Committee on Artificial Intelligence as a representative sample of interest and value to Canadian industry. Abstracts provided are reprinted from "Key Abstracts in Artificial Intelligence" with permission from INSPEC. INSPEC is widely recognised as the leading English-language database covering the published information in the field of physics, electronics and computing. Information contained in the INSPEC services is collected on an international basis from over 4,000 Journals and 1,000 Conference Proceedings. INSPEC is a dividion of the Institution of Electrical Engineers, Station House, NIghtingale Road, Hitchin, Herts, United Kingdom. All INSPOEC's products and services are available in North America from the INSPEC Dept. IEEE Service Centre, 445 Hoes Lane, P.O. Box 1311, Piscataway, N.J. 08855-1331, U.S.A."

Persons wishing to obtain copies of references cited should contact their nearest technical library or the Canada Institute for Scientific and Technical Information National Research Council, Building M-55, Montreal Road, Ottawa, Ontario, CDA, K2A 0-S2, (tel.: (613) 993-1585, Telex 053-03115). For on-line ordering, CAN/OLE users may use the CAN/DOC command. Envoy users type "COMPOSE CISTI."

1816 Applying knowledge-based expert system to meat grading.
Y.R. Chen
(Meat Animal Research Center, USDA-ARS, Clay Center, NE, USA).
Proceedings of the Annual A1 Systems in Government Conference (IEEE Cat. No. 89CH2715-1). Washington, DC, USA, 27-31 March 1989 (Washington, DC, USA: IEEE Comput. Soc. Press 1989) p. 120-3.

The author presents an ongoing project, which has the immediate objective of developing a knowledge-based expert system to assist meat graders and an ultimate objective of automating the meat grading process through applications of image processing, pattern recognition, and expert systems. A voice-input, knowledge-based system for grading carcass beef has been developed and tested. The meat grader provides input for the system in the form of carcass characteristics by speaking into a microphone headset. On reception of the characteristics of the carcass, the knowledge-based system reasons through the production rules to reach quality and yield grades for the carcass. (8 refs.)

1823 An expert system using prioities for solving multiple-criteria facility layout problems.
B. Malakooti, A. Tsurushima
(Dept. of Syst. Eng., Case Western Reserve Univ., Clevelaned, OH, USA).
Int. J. Prod. Res. (UK), vol. 27, no. 5, p., 793-808 (May 1989).

Develops an expert system for multiple-criteria facility layout problems. The facility layout problem is identified as an ill-structured problem: the authors' approach for solving it is based on expert systems and multiple-criteria decision making (MCDM). The expert system interacts with the decision maker (DM), and reflects the DM's preferences in the selection of rules and priorities. The inference engine is a forward-chaining reasoning procedure which is discussed in detail. The approach consists of two parts: (a) construction of a layout based on a set of rules and restrictions, and (b) improvement of the layout based on interaction with decision maker. The MCDM expert system approach considers and incorporates the multiple criteria in these two parts as follows. In (a) it uses priorities on the selection of rules, adjacency of departments, and departments for construction purposes. In (b) it uses different objectives such as materials handling cost, flexibility, and materials handling time for paired comparison of generated layouts for improvement purposes. Some experiments with the developed computer package are reported and an example is solved. (23 refs).

1825 Computer-aided decisions in human services: expert systems and multivariate models.
F. Sicoly
(East York Board of Educ., Toronto, Ont., Canada).
Comput. Hum. Behav. (USA), vol. 5, no. 1, p. 47-60 (1989)

Two approaches to the development of computerized supports for decision making are compared. Expert systems attempt to codify the formal and heuristic knowledge of human experts in the form of rules. However, estimates of relationships and probabilities provided by experts are prone to error and distortion. In contrast, multivariate procedures derive rules and relationships empirically by using the accumulated evidence from hundreds of cases on a data-base. Results from several fields have consistently demonstrated that conclusions generated using statistical models are equal or superior to the decisions made by clinicians. The major strength of expert systems is the use of natural language and explanation facilities which make them more intelligible to the user. Combining this aspect of expert systems with the power of multivariate procedures may allow for the development of an appproach that achieves optimal performance but is also more acceptable and accountable to the user. (39 refs.)

1871 Comparing vector space retrieval with the RUBRIC expert system.
F. Gey, Wingkei Chan
(Dept. of Comput. Soc., Lawrence Berkeley Lab., CA, USA). SIGIR Forum (USA), vol. 23, no. 1-2, p. 5-15 (Fall 1988 - Winter 1989) (received: 10 Apr 1989)

RUBRIC is an expert system for full-text information retrieval. The underlying model for RUBRIC's information retrieval process is based upon fuzzy set theory. The RUBRIC developers have compared RUBRIC to the Boolean retrieval model, which it subsumes. This study compares RUBRIC to the vector space model for information retrieval, using RUBRIC's test collection of thirty news articles from the Reuters News Service and their test search for articles which satisfy the information need to find out about 'violent acts of terrorism'. Results indicate that the vector space model is comparable to RUBRIC for relevant documents, while RUBRIC performs better at retrieving marginally relevant documents. (13 refs.)

1877 Prospector II.
R.B. McCammon
(US Geol. Survey, Reston, VA, USA).
Proceedings of the Annual AI Systems in Government Conference (IEEE Cat. No. 89CH2715-1). Washington, DC, USA. 27-31 March 1989 (Washington, DC, USA: IEEE Comput. Soc., Press 1989), p. 88-92

Prospector II has been designed as a frame-based expert system and the successor to PROSPECTOR to assist the geologist in performing regional mineral resource assessments. The changes that have occurred relate to the role of descriptive mineral deposit models in resource assessment and to the arrival of graphics-oriented workstations. These changes have brought about an expansion of the volunteer mode, the glossary of geological terms, and the explanation facilities in PROSPECTOR. The information volunteered by the geologist during a session with Prospector II is used to identify the deposit model that best fits the observations and to provide an explanation of what is explained or not explained by the model, and to indicate what is missing in the observations with respect to the model. The advice that is offered gives the geologist greater confidence in choosing a model and in deciding what additional data are needed to perform an asseessment in an area. (11 refs.)

1878 The evolution of an expert DSS folr electric utility load research.
K. Muralidhar
(Dept. of Manage., Nebraska Univ., Lincoln, NE, USA)
M. Tretter.
Comput. Oper. Res. (UK), vol. 16, no. 3, p. 247-56 (1989).

Load research conducted by electric utility companies is an activity embracing the measurement and study of electrical load to understand the trends and behavior of electric utility consumption. The results of load research are major determinants in rate design and capactiy planning for the electric utility company. Accurate load reseach is imperative for a well managed electric utiility. The information base used in load research consists of customer billing data and sample data collected under strict guidelines set by the Public Utilities Regulatory Policies Act. Selection of appropriate sampling procedures is a central function of load research. The paper describes the evolution of an integrated expert decision support system (XDSS) for load research sample design. It is an exploratory system that has a potentially large impact on the profitability of electic utility companies. It was the result of a consulting project begun 4 years ago with a medium sized utility company. Much of the underlying statistical sampling methodology in the XDSS had to be newly derived to meet the specific needs of sampling in load research. (20 refs.)

1880 A hierarchical expert system for computer process control.
D. Ionescu, I. Trif
(Dept. of Electr. Eng., Ottawa Univ., Ont., Canada).
Eng. Appl. Artif. Intell. (UK), vol. 1, no. 4, p. 286-302 (Dec. 1988).

This paper reports on the design and implementation of an expert system for computer process control (HESCPC). The complexity of the expertise necessary for computer process control applications requires that the expert system architecture be structured into a hierarchy of classes of specialized experts. The architecture of HESCPC integrates four classes of expert systems: operator/ manager companion expert class, control system algorithm design expert class, hardware expert class, and software expert class. The paper is concerned with design and implementation of the general system architecture, an operator adviser expert for a nuclear power plant and a control system designer expert using a state space feedback approach. Although the design and implementation aspects of all experts are discussed, the emphasis is on the latter. At this stage of the HESCPC development, the declarative knowledge represented by 423 metarules and 1261 rules is distributed on a hierarchical structure among 20 experts on different levels of the hierarchy which are able to communicate among themselves to solve difficult control problems. Examples of control system design sessions of linear mono and multivariable systems using feedback state space approach are given. A run time of an operator-adviser data-driven expert system for a nuclear plant is also presented. (10 refs.)

1892 Automating process planning: using feature interactions to guide search.
C. Hayes
(Carnegie Mellon Univ., Pittsburgh, PA, USA).
J. Manuf. Syst. (USA), vol. 8, no. 1 p. 1 - 15 (1989)

Machinist is an expert system that automatically makes process plans for fabricating metal parts on a CNC machine tool. It is part of an overall effort to automate the job shop. The type of parts it handles are prismatic, with features cut into one or more sides. Parts of this type are difficult because features on different sides may interact with each other: cutting one group of features first may make the part such an odd shape that it is difficult to clamp for subsequent operations. These interactions must be carefully considered when grouping and ordering the machining operations. The machinist program has been designed to be an integral part of CAD systems, facilitating the generation of manufacturing plans. (41 refs.)

1906 A prototype expert system to perform safety evaluation of alternative architectures for offshore platforms.
A. Lancia
(Tecsa Srl, Levate, Italy), L. Scataglini, L. Bortolazzo, R. Romano.
Reliability Data Collection and Use in Risk and Availability Assessment. Proceedings of the 6th EuReDatA Conference, Siena, Italy, 15-17 March 1989 (Berlin, West Germany: Springer-Verlag 1989), p. 246-52

Basic safety evaluations should be carried out before all the key design decisions are taken. This document describes the architecture of an expert system prototype in order to perform safety analysis architectures for offshore gas platforms. (4 refs.)

1917 Expert systems in civil engineering - state of the art.
S.J. Fenves
(Carnegie Mellon Univ., Pittsburgh, PA, USA).
Fourth International Symposium on Robotics and Artificial Intelligence in Building Construction. Haifa, Israel, 22-25 June 1987 (Haifa, Israel: Technion 1988), p. XV-XXXV vol. 1.

The methodologies and concepts of AI, as embodied today in the methodology of knowledge-based expertsystems (BKES), provide an intellectual framework for addressing heuristic problems in civil engineering that could not be successfully approached with conventional programming techniques based on well-structured, algorithmic formulations. The scope and capabilities of the current generation of KBES may prompt the observation that AI in general and KBES in particular have been 'oversold'. Many of the present civil engineering KBES address trivially small problems, and very few KBES are in production use. This situation is to be expected in the early, formative stages of a new methodology. Current KBES development frameworks or shells were not motivated by engineering needs, and the current state of engineering expertise is not compiled in a form immediately suitable for incorporation into the knowledge base of a KBES. The state of KBES today parallels that of algorithmic computing 30 years ago, in terms of both primitive development languages and primitive available knowledge. Even the most straightforward algorithms have improved by many orders of magnitude from the original ones. (35 refs.)

1918 An expert system for construction planning.
C. Hendrickson
(Dept. of Civil Eng., Carnegie Mellon-Univ., Pittsburgh, PA, USA). C. Zozaya-Gorostiza, D. Rehak, E. Baracco-Miller, P. Lim.
Fourth International Symposium on Robotics and Artificial Intelligence in Building Construction, Haifa, Israel, 22-25 June 1987 (Haifa, Israel: Technion 1988), p. 1-26 vol. 1

Among other concerns, construction planning involves the choice of construction technology, the definition of work tasks, the estimation of required resources and durations, the estimation of costs, and the preparation of a project schedule. A prototype knowledge-intensive expert system to accomplish these tasks. COINSTRUCTION PLANEX, is described in this paper. This system generates project activity networks, cost estimates and schedules, including the definition of activities, specification of precedences, selection of appropriate technologies and estimation of durations and costs. The CONSTRUCTION PLANEX system could be useful as an automated assistant in routine planning, as a laboratory for the analysis and evaluation of planning strategies, and as a component for more extensive construction assistance systems involving design, site layout or project control. The current application for CONSTRUCTION PLANEX is to plan modular high-rise buildings, including excavation, foundation and structural construction. (19 refs.)

1946 The use of memory in text processing.
M. Lebowitz
(Morgan Stanley & Co., New York, NY, USA.)
Commun. ACM (USA), vol. 31, no. 12, p. 1483-502 (Dec. 1988). (received: 02 May 1989)

The performance of a natural language processing system should improve as it reads more and more texts. This is true both for systems intended as cognitive models and for practical text processing systems. Permanent long-term memory should be useful during all stages of text understanding. For example, if, while reading a patent abstract about a new disk drive, a system can retrieve information about similar objects from memory, processing should be simplified. However, most natual language programs do not exhibit such learning behavior. The author describes how RESARCHER, a program that reads, remembers and generalizes from patent abstracts, makes use of its automatically generated memory to assist in low-level test processsing, primarily involving disambiguation that could be accomplished no other way. He describes both RESEARCHER's basic understanding methods and the integration of memory access. Included is an extended example of RESEARCH processing a patent abstract by using information about several other abstracts already in memory. (58 refs.)

"Based on the information provided in the abstracts, the references provided have been selected by the secretariat of the NRC Associate Committee on Artificial Intelligence as a representative sample of interest and value to Canadian industry. Abstracts provided are reprinted from "Key Abstracts in Artificial Intelligence" with permisssion from INSPEC. INSPEC is widely recognised as the leading English-language database covering the published information in the field of physics, electronics and computing. Information contained in the INSPEC services is collected on an international basis from over 4,000 Journals and 1,000 Conference Proceedings. INSPEC is a division of the Institution of Electrical Engineers, Station House, Nightingale Road, Hitchin, Herts, United Kingdom. All INSPEC's products and services are available in North America from the INSPEC Dept. IEEE Service Centre, 445 Hoes Lane, P.O. Box 1311, Piscataway, N.J. 08855-1331, U.S.A."

Persons wishing to obtain copies of references cited should contact their nearest technical library or the Canada Institute for Scientific and Technical Information, National Research Council, Building M-55, Montreal Road, Ottawa, Ontario, CDA, K2A, 0S2, (tel.:(613) 993-1585, Telex 053-03115. For on-line ordering, CAN/OLE users may use the CAN/DOC command. Envoy users type "COMPOSE CISTI."

Advertisers Index

CSCSI/SCEIO Application and Order Forms

Use the forms below to subscribe to the journal ***Computational Intelligence***, to order publications and to join the Canadian Society for Computational Studies of Intelligence (CSCSI/SCEIO), with which you will receive ***Canadian Artificial Intelligence***. Complete the form of you choice and send it to CIPS (which administers membership for the CSCSI/SCEIO) at the address below, with the appropriate fee. (NOTE: Those residing outside of Canada who wish to order the 1988 CSCSI/SCEIO Conference Proceedings must mail the order form to Morgan Kaufmann Publishers.)

CIPS, 243 College Street (5th floor), Toronto, Ontario, CANADA M5T 2Y1

CSCSI/SCEIO Membership
Canadian Artificial Intelligence back issues

❑ I wish to join CSCSI/SCEIO and receive Canadian Artificial Intelligence ($35.00 Cdn.).

❑ I am a student (subtract $10.00 Cdn.)
and/or
❑ I am a member of CIPS (subtract $10.00 Cdn.)

❑ Please send me the following back issues of Canadian Artifical Intelligence ($10.00 Cdn. each including postage, $15.00 Cdn. for issue #3).*

Name ______________________________

Mailing Address ______________________________

____

* If an issue you request is out of print, a photocopy will be provided. Issue #3 (March 1985) includes the supplement Towards a Canadian Fifth Generation Research Plan.

Computational Intelligence
Subscriptions

❑ Please enter my subscription to Computational Intelligence at the CSCSI/SCEIO non-institutional member discount rate ($20.00 Cdn.). ‡

Name ______________________________

Mailing Address ______________________________

____

‡ Computational Intelligence subscriptions are filled by its publisher, the National Research Council of Canada. CIPS only certifies your eligibility for the discount and fowards your order.

CSCSI/SCEIO Conference Proceedings

Please send me the following CSCSI/SCEIO conference proceedings (indicate number of copies desired):

❑ # of copies ___ Edmonton, 1988 (Within Canada: members – $35.00 Cdn.; non-members – $40.00 Cdn. Add $5.00 Cdn. for postage. Mail to CIPS.) (Outside Canada: $35.00 US. Postage within U.S.: $2.00 US for the first copy, $.75 US for each additional copy. Postage outside U.S.: $3.00 US for the first copy, $2.00 US for each additional copy. Within California: add 7% sales tax. Mail to: Morgan Kaufmann Publishers, Order Fulfillment Center, P.O. Box 50490, Palo Alto, CA 94303, USA)

❑ # of copies ___ Montréal, 1986 ($30.00 Cdn. Add $5.00 Cdn. postage for within Canada, $7.00 Cdn. for outside Canada. Mail to CIPS.)

❑ # of copies ___ Saskatoon, 1982 ($25.00 Cdn. Add $5.00 Cdn. postage for within Canada, $7.00 Cdn. for outside Canada. Mail to CIPS.)

❑ # of copies ___ Victoria, 1980 ($25.00 Cdn. Add $5.00 Cdn. postage for within Canada, $7.00 Cdn. for outside Canada. Mail to CIPS.)

Name ______________________________

Mailing Address ______________________________

____

Canadian Artificial Intelligence

Intelligence Artificielle au Canada

AI System Development Tools

Soft Warehouse Inc.
mu-Lisp: Lisp language programming environment. Turns a PC into an integrated workstation for developing AI software. $365. With compiler, $490. mu-Math: Symbolic mathematics system. Complete algebra system solves problems of algebra, trigonometry, and calculus. From $305 to $365. Derive: Mathematical assistant for the PC. Solves large and complex problems. $245.

NeuralWorks Professional II
Advanced, easy-to-use "graphic-intuitive" approach results in a seamless interface. User-defined neurodynamics, and loadable control strategies provide complete freedom in designing multi-paradigm networks or developing completely new network types. IBM PC AT or PS-2 and MAC Plus, MAC SE, MAC II $1,495. SUN/3 or SUN/4, $3,655.

Nexpert
Full scale expert system development tool for PCs with rich graphics support. Object hierarchy with multiple inheritance. Rules support pattern matching, integrated forward and backward chaining, automatic goal generation, non-monotonic reasoning. Open architecture allows integration with external programs. For use on IBM AT, 386, PS/2 and Macintosh II, $6,100. For use on DEC VAX 2000, II, III under VMS/UIS, and UNIX workstations, $9,760.

N-NET
Combined neural/knowledge based industrial controller. N-NET™ 210 is a neural net, based on the Functional Link Net architecture, which may be embedded in expert systems. Use of N-NET™ 210 has shown quality improvement, cost reduction and efficient use of time in the areas of trend analysis, forecasting, process control, quality control, product design and analysis and machine diagnostics. Has been used in speech recognition. Supervised and unsupervised learning and associative recall are all accomplished within the same command and data structure. Runs on IBM PC/XT/AT, PS/2, or compatible. From $850. A VAX version is forthcoming.

Savvy PC
Sophisticated applications generator which has a powerful pattern recognition facility. Three major components: the Savvy Database Manager to store, manipulate, and retrieve information in plain-language style; the Savvy Language & Tutorial to allow customization of database applications; and the Savvy Retriever to translate English queries using pattern recognition. Runs on IBM PC or compatibles. $415.

LPA PROLOG Professional
First PROLOG compiler to allow development of large applications on IBM PC AT. Provides a general purpose high-level programming language with a superb range of functions and data handling capabilities. PC Dual Compiler System $1,215. flex (Forward chaining Logical EXpert) system toolkit (ms-dos) $485. MacProlog Color Edition $725.

ACCENT SA
A high quality text-to-speech convertor converts ordinary ASCII text into intelligible speech. Can be plugged directly into any IBM PC, PC/XT, PC AT or compatibles. Can speak in two modes: the text mode and the spell mode. Speech output may be directly connected to a speaker. A standard RS-232C serial port is provided for interface with a computer or terminal as the host. Prices range from $485–$1,760.

Arity Knowledge Systems Development Pack
A new generation AI prototyping tool for PCs. A prolog interpreter/compiler, a rule/frame-based Expert Systems shell with object hierarchy and inheritance, the external file and database access, a screen design kit, linkage to conventional languages, and an IBM SQL access. $1,315.

Smalltalk/V
Pure object oriented programming. Provides access to a rich tool set: bit-mapping graphics, a point-and-click user interface, multiple windows, and a source-level debugger. Runs on IBM PC/XT/AT/PS or compatibles with at least 512K memory and one of the following graphics boards: Hercules. CGA. EGA. VGA, AT&T 6300, Toshiba T3100 or IBM 3270. Add on features include: EGA/VGA Color Extension Pack, Communications Application Pack, and "Goodies" Application Pack. Price ranges from $125–$245.

Dragon Systems, Inc.
Dragon's VoiceScribe and DragonWriter voice recognition systems have been acclaimed the best voice recognizers for IBM XT, AT and 386 machines. They have constantly outperformed systems costing several times more. The design is based on a sound set of algorithms developed by two of the top researchers in the research community, Drs. Jim and Janet Baker. The devices are ideal for such applications as command/control, data entry/retrieval, form filling (hospital, government), documentation creation or dictation, front-ending of expert systems, and support systems for the handicapped. From $1,215 to $10,980.

Delphi Common Lisp (DCL)
Powerful new standard object-oriented programming facility. Compact, portable implementation of the complete Common Lisp standard. "Multithread" facility to create and control concurrent threads of program execution. Binary license for Sun 2, 3/50 and 3/100, $975. Sun 3/60, 3/200, and 386i, $1,945. Sun 4, $2,930. Apollo DN, $1,950.

DENDROS-1G Evaluation Board
All-analog, electronic ASIC and later VLSI implementation of a neural network which learns and categorizes in real-time. The dynamics of learning and sleep can be observed. $850.

ENQUIRY AND ORDER: Call (613) 592-3030 or send Purchase Order to Applied AI Systems, Inc.
Shipping and Handling charges and provincial sales tax (Ontario residents only) apply.

Applied AI Systems, Inc. is an authorized dealer of the above AI software/hardware products.
Prices are subject to change without notice.

Applied AI Systems, Inc.
Suite 602, Gateway Business Park
300 March Road
KANATA, Ontario, Canada K2K 2E2
Tel. (613) 592-3030
Fax (613) 592-2333