INTERFACING CARTOGRAPHIC KNOWLEDGE STRUCTURES AND ROBOTICS

F. Bouillé

Laboratoire d'Informatique des Sciences de la Terre
L.I.S.T., Boîte 1000
Université Pierre et Marie Curie
4, Place Jussieu
75230 Paris cédex 05
France

ABSTRACT

The HBDS model has been used to build a catalog of various knowledge structures, mainly applied to the cartography and connected topics. An HBDS structure is described and processed using the ADT'81, a very high level programming language designed both for scientific computation and artificial intelligence developments. An HBDS structure carries its rules, directly included; its components look like processes, which can be activated or passivated, and can interfere. Such a process has been given properties, owing to selective captors, allowing to consider that these components have senses. Simultaneously, a robot can be modelized likewise. HBDS having its own inference engine, we obtain an intelligent system for an environment handling, giving the robot a basis of what it is exploring. Main potential applications are military and petroleum exploration.

PURPOSES

Many cartographic applications, and surveying operations will be soon performed by robots, with a more or less advanced technology. Through its captors, a robot has a rough knowledge of its environment. To be autonomous, it must have its independant intelligent system, in order to take decisions, reasoning on the known facts by applying some rules, possibly deducing new facts, or building new rules. Thus, the robot needs a portable intelligent system, abble to represent the real world in terms of cartography, and to perform some algorithms (relevant to the geomatics), better, to apply some heuristics. It needs a sophisticated representation of space, and extended capabilities of fast and efficient processing. That is the main purpose of interfacing cartographic knowledge structures, in a Structured Expert System (SES, built around HBDS), and the robotics.

HBDS AND ASSOCIATED TOOLS

HBDS (Hypergraph-Based Data Structure) was designed between years 1975 and 1977, with the main purpose of representing phenomenons of the real world, as complex as they may be , and storing them in a completely hidden very large data base; nevertheless, the user has the impress that these data are all resident in core memory; the HBDS model based on six basical Abstract Data Types is recursive and self-extensible; because it was presented summarized in many conferences,among which cartographic conferences, we will not describe it here, but just send the Reader to some previous papers (Bouillé, 1979, 1982). In association, other tools have been designed, both for brainware and software purposes. First of all, the numerous algorithms are written in the EXEL mathematical and algorithmic language (Arsac, 1974). A programming language has then been designed and developped for Abstract Data Type handling, named ADT'81; it includes the HBDS complete pro - cessing, as well as inference expression and scientific computation. Around HBDS and ADT, a Structured Expert System has been developped , named SES; we will give later further features of this one; a last layer is used for ensuring the dialog between the system and the user, based on a structuring tool, named the Q-graph. HBDS, ADT and SES present original solutions which make cartographic processing easier and more reliable.They are relevant to "brainware" (Kitagawa,1981).

KNOWLEDGE STRUCTURES IN CARTOGRAPHY

Before trying to develop some cartographic algorithms, we firstly must recognize the intrinsic structure and nature of the phenomenons which are possibly mapped, and likewise the structures of the maps produced.

Building the catalog

Any theme has been associated with a structure, as complete as possible, though the exhaustivity is not realistic.
For a complex theme, we may obtain, for instance, 150 classes representing the sets, approximately 1000 attributes representing the properties, and 600 links representing the potential relationships. See for instance (Le Graverend,1980). During the past years, this catalog was elaborated, and the themes independently considered, were progressively connected. We thus may process any combination of these themes, as a substructure of the big one, or as a combination of this catalog of structures.

Methodology of problem solving

On the contrary of what is generaly taught, we must emphasize the fact that the structure is completely independent of the problem we want actually to solve. Moreover, a structure built correctly,and with no "a priori" idea, always contains a substructure which immediately answers our problem, if we suppose this one not completely stupid.

Storing the catalog

The logical storage, completely hidden to the user, is ensured by the HBDS system itself, by tha way of DSS (Data System Strings) built by the system, according to an internal code independent of the computer; owing to this coding technics, these DSS are portable between computers (8- , 16-, 32- or 64-bit memory word computers); the user may thus program its applications without taking into account any consideration of the data format... These cartographic DSS may then be transferred by a teleprocessing network from a given computer to another one, even allowing an heterogeneous distribution of the data base.

GENERAL IDEAS ON EXPERT SYSTEMS

As it is generaly said, an Expert System is composed of two mechanisms connected; a Knowledge Base and an Inference Engine. The Knowledge Base is composed of two bases, one is the Fact Base, another one is the Rule Base. A fact base is like a classical data base, while the Rule Base contains the elements of inductive and/or deductive processing. The Inference Engine, more or less sophisticate may work with two essential ways; firstly, starting from known facts, applying rules, it progressively builts inferred facts; secondly, starting from supposed facts we would like to consider as hypothesis possibly verified, it considers what rules might give these facts, with what facts, and recursively it finds paths to known facts (or does not find any...); if yes, supposed facts become verified facts. Dealing with a possible great amount of paths, this engine generaly uses the "back-tracking" method, which is very slow and expensive by time-cosuming. In most of Expert Systems, time for finding the rules to be applied on facts progressively increases (generaly, it is an exponentialy increasing) with the number of rules which have been stored in the Rule Base.

THE S.E.S., STRUCTURED EXPERT SYSTEM

We do not agree with the distinction between facts and rules; in the SES, components of the knowledge structure may carry algorithms; any cartographic information may be associated with a CONDITION, which is systematicaly tested each time this information is used; likewise, a cartographic information may be associated with an ACTION which is performed each time the information is used; this ACTION and this CONDITION are set and updated possibly at execution time, as simple character strings, which may be assigned or read.

Rule-based cartographic system

Rules are expressed in the ADT programming language; rules may be associated to any facts; in the HBDS system, moreover, there is no major difference between a fact and a rule; a rule in ADT is a test, which condition may be as complex as needed, and the action may contain any programing statement(s); rules can be entered at any time; these rules can deal with the cartographic abstract data types previously expressed in the Knowledge Structure; SES ensures the immediate connection between any "fact" with the only rules which produce and/or consume them; in the same time, in ensures the reciprocal connection between any "rule" and the facts which are produced and/or consumed by it. Moreover, a given rule may immediately activate (the term is "APPLY") one or more other rules. On SES in cartography, see (Bouillé, 1983, 1984).

Deleting the back-tracking concept
The capability of connecting inside the knowledge structure "facts" and "rules" (or so-called) has two important effects: first of all, the time used for retrieve the rules to be apppied does not depend on the number of rules stored into the system. It only depends on the rules to be applied; there is then an order between rules, with possible independency we will discuss at the next paragraph. The second important effect is in the deletion of the back-tracking concept; the way ADT works is without tree-structures, and it provides an economy of time.

Vectorized and parallel processing
When considering a cartographic abstract data type, there are often several rules to be applied, some of them being sequential with a given order, others being independent; thus, they are potentialy parallel; but classical computers are sequential... Sometimes a rule may operate not on one only fact, but on sveral facts of same type; one only type of operation on several data implies a potential vectorization; but it requires, to be effective, a vector processor, and not simply the scalar processor like on the classical old machines distributed on the market. It requires too a special software, first for vectorizing the processing, second for distribute the work among the processors in a parallel context.
ADT'81 has been designed on the basis on an extended EXEL and comprises the capability of vectorization and parallelism both for scientific computation and inference processing on very large computers (Bouillé, 1984); other large projects presently begin on parallel prototypes, which support remote-sensing, cartographic and surveying applications.

INTEGRATING SENSES IN THE STRUCTURE

A human being has different senses (sight, smell, etc). These capabilities may be represented as attributes of the concept "human" in a structure; but any other component can have other (or equivalent) properties, which can be represented too.

The concept of observer
Associated to the existence of senses, we define some types as Observers; they carry attributes related to properties considered as senses. Any property has some limits, is expressed in some units, and works only in a given context, possibly conditionned by some rule(s), possibly when activated by a given process. For cartographic applications, the sight is the most important property, but is not the only one; others have

a part to play, which can be mapped for instance.

The concept of observable object

When having defined a given property as a sense, all the components presenting the opposite property of emitting the corresponding information are said "observable" in correspondance with given observers; of course, an observer may be too observable. These capabilities are directly carried at the level of the knowledge structure. For instance, a cartographer in the landscape is an observer who can be seen by another observer (if this one is not blind, and if ist cartogrphic position, that of the observer and the relief allow the observation; we must consider the direction the observer is looking at...).

Weekness of actual graphical languages

Considering standards like GKS and others, we are very far from what we need for expressing a scene like an observer looking at three-dimensional observable objects in a given window. If a component is graphical and stays in the field I am looking at, to make it appear on a graphic screen, I must write many programming graphical statements. Why? When dealing with abstract data types, it is possible, like in the HBDS, to introduce the concepts allowing to define observers and observable objects; likewise it is possible to define SEE, as a procedure written once and only one time using the graphical property to build one image which is exactly what is seen by the given observer. That is precisely what ADT allows.

INTRODUCING THE ROBOT CARTOGRAPHER

What we have said for a human is good for a robot, excepted this one has not exactly the same senses, is said unintelligent (supposing the human is intelligent...), cannot progress in the landscape like the human being, etc.

The robot as an observer and an observable object

The robot has some special features characterising the way it can works, the senses which are operational and in what context, its position, and the position of the articulated arms, its speed, what organs are on or off, the directions the sensors are working and what it is presently capturing. If this robot has in his decision system a knowledge structure expressing a given mapped environment, if we express its absolute position and direction, this observable object becomes an observer; informations captured are provided to the "FAS" or Fact Acquisition System, which must recognize the nature of these facts and must give them to the knowledge base; other informations may be entered, such as new rules; others may be deduced by the inference mechanism, for giving either new facts , infered , or new rules bylearning; facts provides from multi-sensors (Talou, 1981).

Architecture of the"robot cartographer"

As drawn on the figure, there are two main parts, one for the SES, one for the robot; the special feature of a cartogrpher realized as a robot is not in the robot but in the intelligent mechanism; moreover, inside the SES, the inference engine is not characteristic at all of the cartography; only facts and rules are specific to the discipline.

The SES comprises a Knowledge Base, noted KB, in which the Inference Engine (IE) is included; this feature is typical of SES; this KB is connected to a Data Base DB, made of two parts, the Logical Data Base (ensured by the HBDS system), LDB, and the Physical Data Base (PDB) depending on the hardware parameters of the machine.

The Robot has specific captors, better named sensors (S), a Power Source, PS, a Transmission System, TS , and a Motion Mechanism, MM.

The Interface works by two ways, one from Robot to KB, one from IE to the Robot, respectively named Fact Acquisition System (FAS) and Action Decision System (ADS).

The FAS is connected to the knowledge base by a subset of acquisition routines, and the ADS is connected to the MS by a subset of action routines.

Modeling the robot and its environment

The robot is represented by some abstract data types ensured by the same system which is used for representing the complete environment; we think that it makes the modeling easier, for the linking between the robot and the environment.

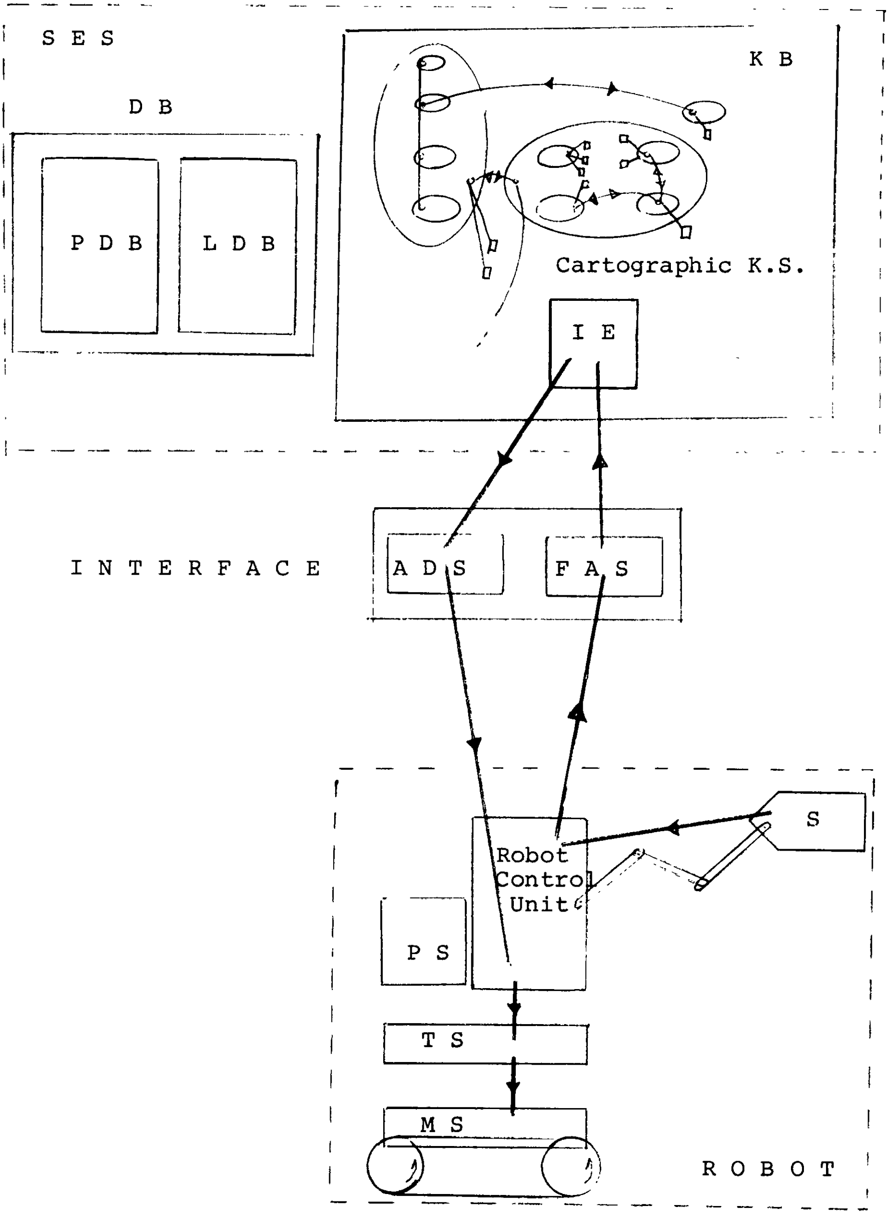
When moving, the robot must take care of the environment, in order to know where it must go, the slope, the praticability; for performing an action, it must know its components, but these comonents, like an arm, for instance, have sometimes consequences on the environment, by modifying the position of some objects; on a slope, the "observer" is not horizontal, of course, and the direction it "looks" at is modified; thus the scene it "sees" is modified; either we must inform the knowledge base of this new direction for repositionning the scene, or we must inform the "observer" he has to modify the position and direction of the sensor, and/or the arm working in the environment.

Thus there are relationships between the observer and the observed ontext. The best way to realize these relationships is to implement the links in the same model and not to connect two different models.

Modeling the fourth dimension

Programming a robot requires to express some contrainsts, and the synchronization between some elementary processes, possibly a process waiting for another one, or the possible parallelism between several processes. This synchronism is inside the robotic system.

There is another synchronism which simultaneously concerns the robot and ist environment, when an action must be activ-



ARCHITECTURE OF THE "ROBOT CARTOGRAPHER"

ated only when something has been modified in the environment or when a process has been activated, hold or passivated in this one.
ADT contains some tools allowing to express the processes inside the Knowledge Structure, and to synchronize them. The conditions of synchronism may be rules, without any difference, the rules being expressed in ADT too.

APPLICATION FIELDS

There are short-dated and medium-dated applications; among the short-dated ones, there is the possibly of improving the model of environment surrounding a robot by using cartographic concepts which are "explained" to this one; there is a chance too, to improve the heuristic and algorithms of the SES in the cartography and spacial field.
As a medium-dated application, there is the possibility of improving the technics of surveying and mapping, making the same work faster and chipper, and to do that where there is a hard environment (radioactivity, toxic gaz, hot atmosphere, undersee), or with senses which are not that of a human cartographer.
The two immediate fields are the military application, and teh oil exploration. In both cases, time is a major factor, and the context is often an hostil environment.

CONCLUSION

We would like to emphasize several points which appear very important for the future of the cartography and the surveying technics.
Whether we agree or we do not, robotics in cartography will increase its part to play. If we want to improve the results, we must improve the technics and methods. Our purpose is not to modify the robot itself, but to adapt the robot to the cartography by upgrading the intelligent mechanism, both the knowledge base and the inference engine.
To obtain a better apprehension of the real world, in connection to the robot, we must express the modeling of both and of their linking with the same model. Moreover, scientific computation, inference processing, and robot commands must be expressed in the same programming language, and this one must work on the absttact data types belonging to the large knowledge structure. Such a system must include capabilities of parallelism and vector processing. Our developments have been done in the HBDS/ADT system which provides all these features.
Finaly, such a connection presents simultaneously advantages in three areas; for the robotics, by increasing the potential power of the robots; for the artificial intelligence, by providing a complex field (the cartography) allowing to test the expert systems; more important for the cartographers, the association of the both tools will increase the capab-

ilities of the cartographic systems, by providing self-moving intelligent systems capturing, storing and using some rules for improving this work.

REFERENCES

Arsac, J. (1974) The Exel Language. I.P. Paris. 64p.

Bouillé, F. (1979) The HBDS and its application to data structuring and complex system modelling. Report US Air Force , 167p.

Bouillé, F. (1982) Actual tools for the cartography today. Euro-Carto I. Cartography and Computer Technology, 27-32.

Bouillé, F. (1983) A structured expert system for cartography based on the HBDS. Auto-Carto VI. 9p.

Bouillé, F. (1984) Parallelism and vectorization with ADT'81 on the Cyber 205. CDC France preliminary report. 7p.

Bouillé, F. (1984) Inference parallel processing in a data base on a Cray-1. CCVR Preliminary report. 4p.

Bouillé, F. (1984) A structured expert system for geodata banking. The role of data in scientific progress. Elsevier, 417-420.

Bouillé, F. (1984) Architecture of a geographic structured expert system. Zürich Int.Sem.on Spatial Data Handl.Proceed. 24p.

Kitagawa, T. (1981) Datalogy and semiology in establishing Knowledge Information Processing Systems in Research. 2nd Int.HBDS Seminar Proceed. 1-24.

Kitagawa, T. (1981) Generalized relative ecospheres of Knowledge Information Processing Systems and their implications for scientific re - search. 2nd Int.HBDS Seminar Proceed. 25-42.

Le Graverend, F.(1980) Cartography and urban planning: a tool for decision-makers. 10thICA-WG-TDSCM Proceed. 8p.

Ta lou, J.C., Briot, M., Bauzil, G. (1981) The multi-sensors which help a mobile robot find its place . Sensor Review.