AN EXPERT SYSIEM FOR INDICATIONS & WARNING ANALYSIS

Douglas B. Lenat — Stanford U. and Teknowledge, Palo Alto, Ca.
Albert Clarkson — ESL, Sunnyvale, Ca.
Garo Kircmidjian — ESL, Sunnyvale, Ca.

ABSTRACT

An expert system was constructed to aid the military intelligence analyst in performing the Indications & Warning task: assimilating hundreds of incoming reports, and predicting where and when an armed conflict might erupt next The system currently contains 60 condition/action rules and 170 other frames that deal with the sorts of objects and processes that are being reported on. It employs a two-dimensional Blackboard to accomodate reports from very different sources, to efficiently trigger relevant rules, and to keep the human analyst abreast of the situation. In the process of building this system, and testing it with professional analysts,* we were led to some nonstandard design decisions which may be of general AI interest:
(1) each rule has strong and weak conditions, which are run in separate worlds. By examining which *important,* high-level conclusions differ, it is possible to pinpoint which few specific facts should be doublechecked (i.e., facts whose certainty the system is very sensitive to);
(2) each rule is represented as a frame, facilitating browsing through the rules, adding new rules, and assigning credit and blame to rules.

I. The I&W Task

Intelligence analysts are charged with monitoring an area (say, the two fictitious countries Upper and Lower X), and must alert their superiors to any forthcoming outbreak of hostilities in that area. In particular, the analyst must explain where and when he predicts an attack on Lower X will originate, and what specific evidence he has to support that expectation. This task is referred to as the Indications -and Warning (I&W) problem. To solve it, the analyst reads hundreds of incoming reports each day from various sources (media articles, broadcasts, reconnaisance reports, human operative repoits, etc.) and maintains a detailed model of what is happening in his assigned countries. For more information, see [Wohlstetter 62], [Belden 77], and [Clarkson 81].

The I&W task was deemed one of the most significant problems of national security by former US Secretary of Defense Harold Brown, in a visit to ESL in the summer of 1982. The overwhelming array of data confronting an analyst suggested a man-machine system to support him. An expert systems approach was called for because (i) the analysts themselves describe their reasoning in informal rules of thumb; (ii) experts for the task are clearly recognized nnd were accessible to us; (iii) expertise for this task is often not present where and when it is needed, due to the high job turnover rate among US defense analysts, and due to the relatively small number of such analysts compared to the number of trouble spots in today's world; (iv) analysts must justify their predictions with a line of reasoning, which is easily available in rule-based expert systems.

II. Design of an Expert System for the I&W Task

Working in conjunction with current and former I&W analysts, we have built an expert system to aid in this task. The various kinds of incoming reports are posted on a Blackboard [Erman *et al* 75] (see Fig. 1), whose dimensions are time (marked off in days) and these three levels of abstraction:

(i) specific reports: "expect a report of 200 troops assembling at Area803 two days from now"
(ii) higher-level indicators: "Ground forces are massing at embarkation points all over Upper X"
(iii) very general states: "Upper X's military forces are now at full readiness"

Sixty rules react to changes on the Blackboard by drawing conclusions and making predictions; i.e., the system "works forward". A new report enters and is recorded on the Blackboard, various rules fire, they cause additional enuies to be added, etc. Eventually this process dies down, and the next report is read. A typical rule says:

R1: If a barracks is suddenly reported to be empty,
Then expect a report, within a day, of (those) troops moving by rail or road to a nearby staging area.

When a BarracksEmpty report is posted on the Blackboard, this rule fires and synthesizes a new TroopsMoving report, and posts it on the Blackboard farther to the right (to indicate it's to occur one day later in time), with a tag indicating that it's only a prediction, not (yet!) an actual incoming report

Rule RI takes incoming reports and predicts future ones. Similar rules exist which predict *past* events that were not recorded; this is important, since intelligence gathering is quite incomplete. For instance, the inverse to the former rule says that if troops are moving by rail, they must have come from somewhere, and a few days earlier some barracks probably emptied out. It is important to make and record these pas/dictions, as they sometimes cause additional rules to fire which in turn make predictions reaching even farther ahead in time than the current day. If we were to redesign the system, aware of the frequency of these rule&inverse pairs, we would choose only state the single *causal* relationship, and have both Rl and its inverse derived automatically from that.

Other types of rules add entries at a higher level of abstraction, namely the indicator level rather than the incoming report level. A typical such rule, which adds one high-level entry to the Blackboard and several low-level ones as well, says:

R2: IF there have been, within a 4 day period, at least 6 reports of troops readying to leave their usual base,
and at least 5 are real reports (not just predictions),
and in at least 4 the number of troops exceeds 500
and there is also an air defense system with them,
and CivilianSupport&Preparedness indicator is active,
Then activate the GroundForcesAssembling indicator,
and keep it active for at least the next 6 days,
and add predicted reports of troops moving in those areas (for each of the *next* 4 days)

It's often quite important to know when an indicator *stops* being active, and thus there are rules whose Then-parts *deactivate* indicators, or predict such deactivation events.

III. AI Issues

So far, this design sounds like a rather standard expert system [Feigenbaum 77; Hayes-Roth *et al* 83]. The program has, however, a few nonstandard features of interest to the AI community:

Representation. Knowledge is represented as frame-like assemblages of attribute/value pairs. The program has separate frames for each abstract indicator (UnusualNavalActivity), each *type* of incoming report (LCU'sAbsent), and -- as actual reports filter in - the program creates a new frame for each such report (LCU'sAbsent00081). In general, a frame is created for each entry made on the Blackboard.

Most reports are of *events,* and each event is usually just a sub-process in an easily recognizable larger process. Thus, emptying barracks is a subprocess of troop movement, which in turn is a subprocess of ground force readiness, which is a subprocess of military readiness for attack, etc. Each frame for a type of report records what it might be a subprocess of, how long it takes, what subprocesses it entails, which of those can be done in parallel, etc. From this information, estimates can be derived of the minimal time to various states (such as full readiness). Knowing what processes are going on, and how soon various states might be achieved, "solves" the

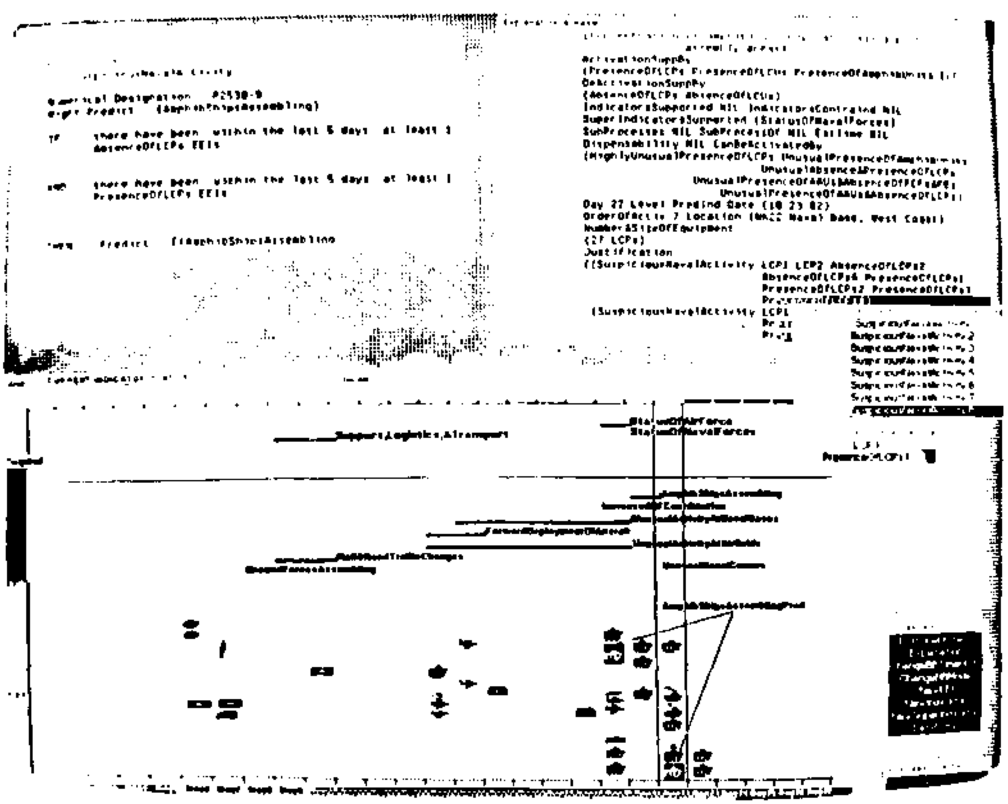

Figure 1. The appearance of the primary Dolphin display screen after the I&W program ran on three weeks worth of incoming reports.

I&W problem. The analyst reads off, at as high a level as desired, predictions of the content and date of future reports.

Besides the types of frames already described, there is a hierarchical network of frames for objects mentioned in reports. This is necesary so that, e.g., a Rile that specifies 'tracked vehicles' will trigger if the incoming report mentions 'tanks'.

A separate frame also exists for each rule. For instance, R2 has not only an If and Then slot, but a Name, a NumericalDesignator, a Creator, a CreationDate, a MightActivate, an IsA, a CertaintyOfConclusion, an Overall Worth, and values for several other slots. This facilitates browsing through the rules, adding new rules, debugging a faulty set of rules, finding related rules when a rule is "almost" relevant, assigning credit and blame to rules, and (hence) automatically modifying the If and Then parts of rules, their certainties, and their worths.

Control. An experimental aspect of the system is to provide rules with two different triggering conditions, an IfStrong and an If Weak. Usually, the former conditions imply the latter, though this is not essential to the idea, and in some cases in our system it is *not* what the experts told us; for instance, one rule's IfStrong has a conjunct of the form "...seen 6 reports in 6 days", and its IfWeak has a corresponding conjunct that tests "...seen 2 reports in 1 day" - neither of these implies the other.

Two separate blackboards are in effect maintained, two entire runs of the program, if you will. In one case, Riles' conditions are assumed to be their If Strong's; in the other case, their IfWeak's. Naturally, this often leads to discrepancies of which rules fire, hence which predictions get made. If the "true" conditions lie somewhere between the IfStrong and IfWeak, then the "true" best predictions ought to be a superset of the IfStrong world's Blackboard, and a subset of the IfWeak world's Blackboard. In particular, if both of them make the same high-level prediction at nearly the same time, it is more likely to be correct. Conversely, if there is a crucial difference between the blackboards, we can list the rules whose difference in IfWeak and IfStrong conditions led to that discrepancy. The specific conditions can then serve to focus our attention on *exactly* what detailed data we should gather in the immediate future; this data is handed to collection tasking planners, whose job is to close such "information gaps", and is no longer part of the I&W problem.

The important point to make here is that, even if there is a wide difference between the low-level predictions that are made on the two blackboards, *only those differences which led to important high-level differences are worth paying attention to.* For instance, in a typical run, the UnusualActivityAtNavalBases and UnusualActivityAtAirfields indicators triggered on different days and in different orders in the IfStrong and IfWeak cases, but still they were both close enough to each other in time that in each case the general Air&Naval Readiness state was highlighted at the topmost level.

Explanation. The analyst must support his predictions, both qualitatively and quantitatively. As our program runs, the user can point (using a mouse-driven cursor) to an item posted anywhere on the Blackboard. S/He then receives information on it, including (if it is not a primitive report coming in as raw data) a listing of the rules that led to its posting. The user gets a stylized English translation of those rules, and can inspect them in detail if s/he desires. By pointing to any clause in the rule's If part, the particular reports (or predictions or indicator activations) that were used to satisfy that clause are highlighted down below, on the display of the Blackboard. The *quantitative* support the analyst requires is calculated from the times taken by all the subprocesses an event requires.

Interface. The program was written in Interlisp-D, and runs on all Xerox D (1100 series) machines. Even on a Dolphin (1100) itself, reports are handled so rapidly (30/minute) that we were forced to artifically slow the program down for demo purposes. The screen is divided into several windows (see Fig. 1), the largest of which is a display of the current state of the Blackboards. The x-axis represents time, the y-axis level of abstraction and certainty. Other windows are used to display: the frame representing each incoming report; a running commentary by the program of what it's reasoning about; menus to customize the display or access the explanation facility; a tree of processes and subprocesses from which readiness time estimates are derived; and finally a list of information gaps (critical pieces of missing or uncertain information, as determined by running in two separate worlds, using Strong and Weak versions of the rules). A separate color screen is maintained, with a map of Upper and Lower X on it As each entry is made on the blackboard, it is also drawn on the map, at its proper location. Figure 2 shows this map display, and Figure 1 the primary 1100 screen, during one run.

IV. Conclusions & Future Directions

The small (50-rule) system we built is little more than a prototype system for aiding an intelligence analyst with the I&W task. Although it is incomplete in some respects, it has served as a concrete example to stimulate analysts into giving many comments for the next generation system for this task. From an AI point of view, the way we overcame some obstacles in this problem may prove to be general methods to add to the Knowledge Engineer's toolkit. We are referring to the flexibility of what the user sees, the representation of rules as frames, and the use of two Blackboards (driven respectively by IfStrong and IfWeak conditions on rules) to find the most sensitive details in the reports, and hence to guide collection tasking.

Two directions are being pursued to continue this work in 1983. (1) The existing system is being expanded, to cover a broader range of report types. (2) We are using techniques of open-ended exploration [Lenat 83] to perform *scenario generation,* based on the same knowledge base of facts and rules used in the I&W

system. The goal of that project is to produce long chains of cause and effect which terminate in particularly undesirable results, and then suggest specific information to gather which, quite early on, could detect such a scenario being attempted.

ACKNOWLEDGEMENTS

This work was pursued at ESL, Inc., funded through internal R&D funds. Ed Richardson, Paul Clarke, and Len Karpf participated in the project along with the authors, and were largely responsible for the explanation facility and the map. Ernie Chaves provided useful management guidance. Consultants from Teknowledgc, Inc. and analysts from government intelligence agencies were also crucial to its success. In particular, we thank Penny Nii and Mike Mitani.

REFERENCES

[1] Barr, Avron, and Edward A. Fcigcnbaum, eds., *The Handbook of AI,* William Kaufman Pub., 1982.

[2] Belden, Thomas G., "Indications, Warning, and Crisis Operations", *International Studies Quarterly,* 21, 1, March, 1977, pp. 181-198.

[3] Clarkson, Albert, *Toward Effective Strategic Analysis,* Westview Press, 1981.

[4] Hayes-Roth, Frederick, Don Waterman, and Douglas Lenat (eds.), *Building Expert Systems,* Addison-Wesley, 1983.

[5] Lenat, Douglas B., "Eurisko: A program that learns new heuristics and domain concepts: The Nature of Heuristics III", *A.I. Journal,* March, 1983.

[6] Erman, Lee D., and V. R. Lesser, "A Multi-Level Organization for Problem Solving, Using Many, Diverse, Cooperating Sources of Knowledge", *Proa 4th IJCAI,* Tbilisi, USSR, 1975, pp. 483-490.

[7] Wohlstetter, Roberta, *Pearl Harbor: Warning and Decision,* Stanford University Press, 1962.

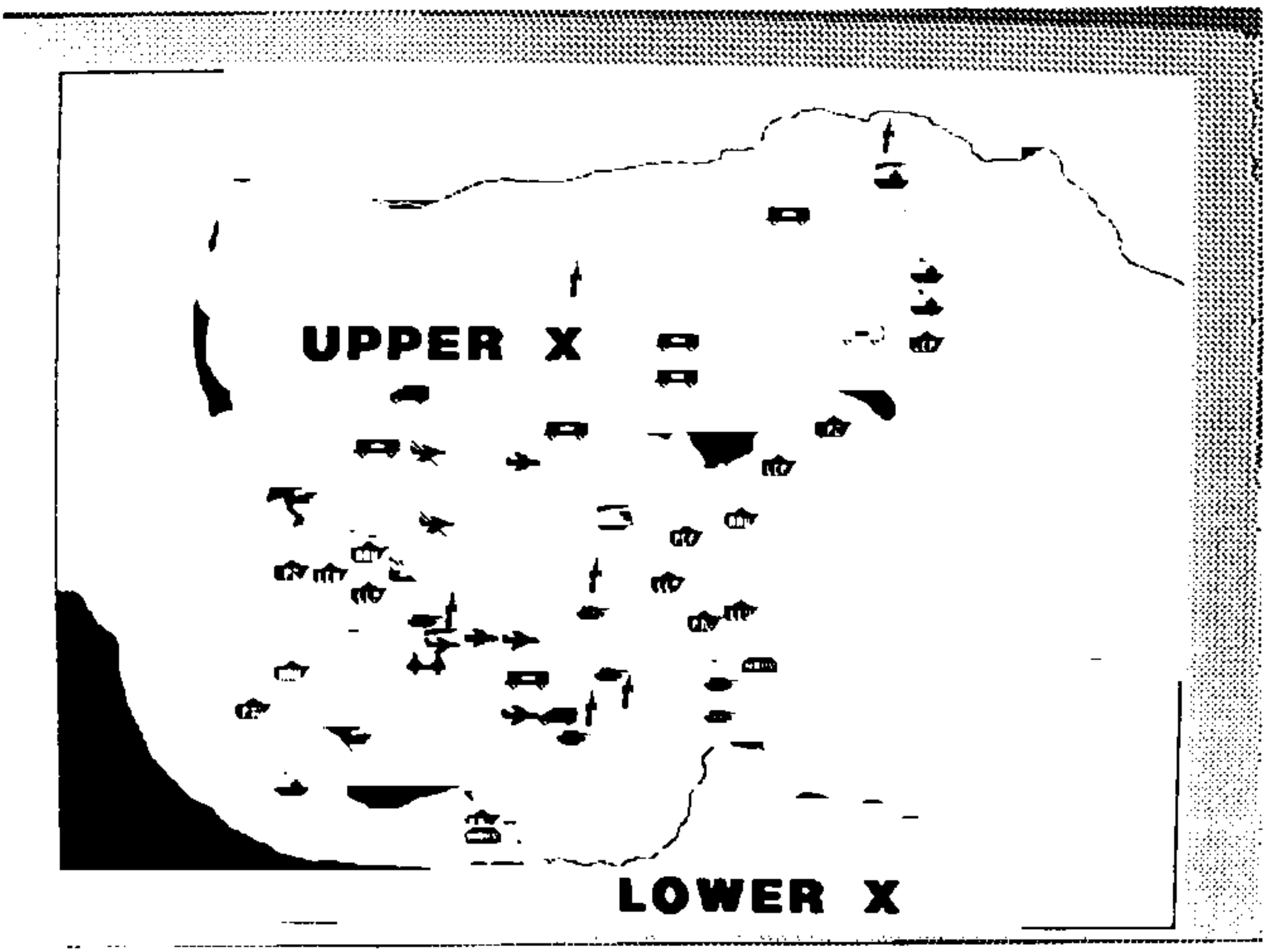

Figure 2. The appearance of the Dolphin color display screen after the same run of the I&W program.