Report no. changed (Mar 2006): SR-123-UU

SACLANTCEN REPORT
serial no.: SR-123

SACLANT ASW RESEARCH CENTRE REPORT

EVA – An Evaluator of Vague evidence in ASW situations:

An expert system prototype

W. Seiche

October 1987

Report no. changed (Mar 2006): SR-123-UU

Page count for SR-123
(excluding covers)

Pages	Total
i–iv	4
1–21	21
@1–@23	23
	48

SACLANT ASW Research Centre
Viale San Bartolomeo 400
19026 San Bartolomeo (SP), Italy

tel: 0187 540 111
telex: 271148 SACENT I

NORTH ATLANTIC TREATY ORGANIZATION

Report no. changed (Mar 2006): SR-123-UU

EVA – An Evaluator of Vague evidence in ASW situations:

An expert system prototype

W. Seiche

The content of this document pertains to work performed under Project 06 of the SACLANTCEN Programme of Work. The document has been approved for release by The Director, SACLANTCEN.

Ralph R. Goodman

Ralph R. Goodman
Director

Report no. changed (Mar 2006): SR-123-UU

intentionally blank page

EVA – An Evaluator of Vague evidence in ASW situations:

An expert system prototype

W. Seiche

Abstract: This paper presents an example of how an expert system can contribute to the creation and updating of an ASW tactical picture to meet the needs of a high-level shore commander. The basic assumptions are a large area of interest and a considerable number of targets in combination with a rather low number of data per individual target and an associated low data rate, resulting ultimately in fuzzy target evaluation and target tracking. The inputs to the expert system model are facts and existence (which may be uncertain, imprecise, or even mutually inconsistent); outputs are appropriate conclusions.

Keywords: antisubmarine warfare ∘ ASW ∘ command and control ∘ EVA ∘ expert system ∘ fuzzy target evaluation ∘ Mediterranean ∘ tactics ∘ target tracking

Contents

1. Introduction

Creation and updating of a tactical picture from the ASW situation in the Mediterranean Sea is a particularly tricky task if this picture is to meet the needs of a high-level shore commander. On the one hand the area of interest is so large that a considerable number of targets can be assumed to be present; on the other hand the amount of data and the data rate originated by an individual target are very low in comparison with those for other tracking problems. In particular, the datum-to-target correlation becomes a problem when the data rate is low with respect to possible target movements. In consequence, the data set from which a target's evaluation has to be derived is not well defined and so both target evaluation and target tracking become fuzzy.

Whereas an increasing number of contact reports would require computer support for updating the tactical picture, conventional tracking algorithms will be jeopardized by the low amount and rate of data for a single target. Conventional computer support suffers also from the heterogeneity of data sets: there are reports to consider which usually originate from a wide variety of sensors and sources. SACLANTCEN therefore decided to look for solutions in the field of expert systems. A subgoal was to acquire experience in this up-and-coming field.

For our purposes, expert system means a computer model which comprises the inferential capability of an expert as well as his knowledge about the state of his world. Inputs to the model are facts and evidence (which may be uncertain, imprecise or even mutually inconsistent); outputs are appropriate conclusions.

The scope of this paper is to give an example of how expert systems can perform in the above task. A particular prototype model, called EVA, is presented and the intention is to encourage the ASW community either to extend the model for in-the-field use or to create its own expert system.

2. Problem description and definitions

Sometimes designations have different meanings within the contexts of different domains, such as, for example, ASW and AI. This paper deals with both of these domains and consequently some definitions have to be given in order to avoid confusion:

Evidence: A fact that gives a reason for believing in the presence of a target, e.g. the content of a contact report.

Evaluation: The categorization of a target or contact according to the evidence. It is similar to identification in ASW terminology. In particular, to evaluate means in this context to correlate a contact or evidence for a contact to a node of the taxonomy tree (Fig. 1—the term EVA in the figure is the name of our prototype expert system).

Level of evaluation: Precision of an evaluation's category designation. To *raise* the level of evaluation means to increase precision (not reliability). The *highest* level of evaluation is the correlation of contacts to one of the tops of the taxonomy tree.

Degree: The extent to which something is manifest. Used here only as degree of belief which indicates how reliable a given evaluation is similar to classification in ASW terminology.

2.1. TRACKING AND DATUM-TO-TARGET CORRELATION

Task: Incoming contact reports and other evidence have to be correlated to individual targets. Specifically, the targets' tracks have to be derived.

The set of reported contacts does not necessarily correspond to the set of detected targets. Some contact reports may be caused by the same target, without that fact being known. Also, some contact reports can be false alarms. (False alarm here means a report stating the presence of a target when there is none.)

Therefore in the worst case, where there is no information about any correlation, each possible combination of contacts has to be regarded as a candidate for a target track. From here on, such hypothetical target tracks are called *clusters*.

All contacts comprising a cluster will transfer their attributes to the cluster (information of first order). Another source of information on clusters is the grouping of contacts: e.g. from two contacts that have no attributes but do have location the evaluation nuclear can be derived just by combining them into a cluster and calculating the transition speed from one to the other (information of second order).

A tactical picture can be regarded as a collection of clusters such that all contacts are contained once and only once in this collection. Usually there will be a lot of possibilities to select clusters for forming a tactical picture. The expert has to choose the most likely one from these possibilities and to present it to the commander as the actual *subsurface picture* (SSP).

Example 1 illustrates the extent of this problem. Example 1 shows how unmanageable the selection of the SSP may become even with a small number of contacts. Careful evaluation of contact and cluster properties is necessary in order to rule out inconsistent clusters (cluster pruning) and so reduce the number of tactical pictures that is possible.

2.2. EVIDENCE POOLING AND EVALUATION OF CONTACTS AND CLUSTERS

> Task: *All available information concerning an individual contact or cluster has to be considered to raise its level of evaluation as high as possible.*

Within this paper information concerning a contact is called *evidence* An example for which kind of target attributes [1] evidence can be obtained is shown in Fig. 2.

Usually an expert will not be completely sure of his evaluation, be it because evidence is not absolutely reliable (possibility of false alarm) or because the supplied information is incomplete. Consequently evaluations will often be below top level (e.g. diesel where something like Tango, Kilo, etc. is expected), ambiguous (e.g. Juliett or Echo II), or believable only to a limited degree. For a proper assessment this degree is a fundamental attribute of any evaluation.

For processing reasons it is useful to express the latter as a *degree of belief* (DoB) with a numerical value, e.g. a percentage, where 100% means the expert is absolutely certain of his evaluation and 0 means absolutely ignorant; disbelief is expressed as DoB in the evaluation's complement. [2]

---

[1] In the software package used for developing our prototype expert system terminology attributes are called Own slot or Member slot. The software package is known as KEE

[2] The DoB is a judgement, i.e. subjective and must not be mixed up with the objective chance that a target of the evaluated type is present. The chance does not depend on the expert's state of information; his DoB clearly does. Of course, if an expert should know the chance, he would take its values as his DoB. Unfortunately probability is often used as a synonym for both.

Example 1

The contacts U, V and W are reported consecutively. There is no information available whether these contacts belong to one or more targets. Let X, Y and Z be the names of hypothetical targets. Then we have to regard

Cluster	As track of target	Based on n reports
C1 [U]	X	1
C2 [V]	Y	1
C3 [W]	Z	1
C4 [U, V]	X[1]	2
C5 [U, W]	X[1,2]	2
C6 [V, W]	Y[1]	2
C7 [U, V, W]	X[1]	3

[1] There is no a priori information about the presence of targets; permutations of clusters and target names can therefore be omitted.
[2] As X but different.

These clusters allow the following tactical pictures:

Picture	Composed of
P1	[C1, C2, C3]
P2	[C1, C6]
P3	[C2, C5]
P4	[C3, C4]
P5	[C7]

If false alarms have to be taken into consideration, additional degenerated pictures are possible:

Picture[1]	Composed of
P1a	[C2, C3]
P1b	[C1, C3]
P1c	[C1, C2]
P1ab	[C3]
P1ac	[C2]
P1bc	[C1]
P1abc	[empty]
P2a	[C6]
P3b	[C5]
P4c	[C4]

[1] Lowercase letters in the picture designation indicate which contact is considered as a false alarm.

Example 2

An expert receives two reports which refer to the same contact. Report A states that there was an interception from a Snoop Slab radar. Report B states that sonar sensors were placed within the assumed area of probability, but no turbine noise was heard.

Through his expertise, the expert evaluates the evidences as follows:

Evidence	Evaluation	DoB
A	Juliett or Echo II	80%
B	not nuclear	60%

Example 3

The expert is asked for his DoB of whether the contact is a Juliett or an Echo II. This example is a table of some different evaluations, showing some peculiarities of the DoB.

Contact	Evaluation	DoB
A[1]	diesel	80%
	not diesel	20%
B[2]	diesel	80%
	false alarm	20%
C[3]	Kilo or Tango	80%
	other diesel	20%
D[4]	Juliett	70%
	Echo II	20%
E[5]	diesel	90%
	Juliett	60%

[1] Contact A is evaluated in a conventional Bayesian manner, where the complement of the DoB is committed to the complement of the evaluation.

[2] The DoB in the evaluation of contact B is also distributed over more than one hypothesis, but independent.

[3] The evaluation of contact C shows a similar distribution, but the hypotheses are disjunctions of nodes of the taxonomy tree (Fig. 1).

[4] In the case of contact D an amount of less than 100% is committed to the DoBs. This indicates that there is a limited belief in the mentioned hypotheses, but no justification for any distribution of the rest. To commit this rest, e.g. to something else, is as plausible as committing it to the mentioned hypotheses.

[5] The evaluation of contact E contains a refinement. In this case the sum of the DoBs can exceed 100%, because the refinement hypothesis implies the coarse one.

If more information arrives, i.e. new evidence for a contact or a new contact to be added to a cluster, the expert will use it to

- raise the level
- resolve ambiguity
- increase his DoB

of his evaluation by creating a synoptic view of all available information and by applying good reasoning. This performance is called *evidence pooling* and its result is the current best hypothesis (CBH) of the contact's or cluster's evaluation.

These problems are illustrated in Examples 2 and 3. Example 2 has shown how the evidence pooling must be performed for the evaluation task, while Example 3 has given an impression of how carefully the notion DoB has to be used to avoid the mix-up of interpretations. It is also clear now why it is important to distinguish DoB from chance: while chance deals only with well defined outcomings (an event happens or does not happen), the DoB admits also vagueness and will then require special treatment.

3. Performance of the EVA model

Our prototype system (EVA) is implemented on a Texas Instruments EXPLORER workstation. The software package used is KEE from Intellicorp Corp. KEE allows frame-based modelling as well as near-plain-language production rule formulation. A user-defined LISP function can easily be attached.

The model contains facts of the ASW world, like hierarchy and properties of target classes, environmental conditions, sensor properties, etc. In other words all those things an ASW expert should be familiar with. Further, it contains rules which describe the interaction of facts in an `if...then...` manner, i.e. it mirrors the know-how of an expert. If evidence for some target appears, the model applies these rules to deduce what kind of target might be present and what it is doing. This procedure is supported by appropriate algorithms, e.g. for evidence pooling, cluster pruning or track selection.

3.1. DESCRIPTION OF ITEMS IN EVA

Figure 3 shows all items of the model available to the user. The rectangles are called windows in KEE terminology.

- The Output window depicts a graph of all those items which contain information for the user. This information can be displayed by requests in that same window.
- The Lisp Listener window provides services such as invoking user-defined functions.
- The KEE Typescript window is the major dialogue interface.

The three little windows on top are merely triggers. If a cursor is moved on one of them, the activity mentioned in its title is executed.

- The New Evidence Creation function serves for supplying the model with new information.
- The Track Assessor's SSP Proposal function proposes clusters which are likely to be real tracks and combines them in the SSP.
- The Dempster/Shafer Inquiry function is a means for the user to ask the system for its DoB in a user-supplied evaluation hypothesis for a contact or a cluster.

Figure 3 is a snapshot of the situation after two evidences (**EVIDENCE.1** and **EVIDENCE.2**) [3] became available:

Typescript window: This display shows the dialogue while inserting the first two evidences. System prompts and responses are in uppercase letters, the user's answers with lowercase initials. Because the user could'nt specify reference contacts, the system created a contact for each evidence (responses **CONTACT.1** and **CONTACT.2**).

Output window: The heart of the knowledge-base graph is the taxonomy tree. The system's evaluation of contacts is displayed as connections (dashed lines) between contacts and the appropriate nodes of the taxonomy tree.

Up to now the system could not find any inconsistency between the two contacts and so it had to form all possible clusters (**CL.1, CL.2, CL.3**) [footnote [3]] for further consideration. The correlation of contacts and clusters is displayed by solid lines. (The distinction dashed/solid line is merely for internal reasons.)

Besides the major features of the taxonomy tree, evidences list and cluster list, the knowledge-base graph contains a representation of the hierarchy of implemented rules, and the items sensor and environment, both intended for future extensions of the knowledge base.

Figures 4 to 8 show attributes and their values for the items **CONTACT.1**, **CL.3**, **ENVIRONMENT** and **SENSOR**. The attributes *Prior Hypothesis* and *CBH* of contacts and clusters need some explanation.

Generally *Prior Hypothesis* represents the distribution of DoB over all the evaluations for which there was an evidence so far. In the case of **CONTACT.1** this distribution was derived by evaluation rules (for rules see Subsect. 5.4) dealing with *signals* (*'A'*) and signal-derived attributes (Snoop Tray). The *Prior Hypothesis* of **CL.3** was derived in two steps: first the *Prior Hypotheses* of the two member contacts were combined (information of first order); then the transition speed from **CONTACT.1** to **CONTACT.2** was calculated and an evaluation rule dealing with speed was applied to it. The conclusions of that rule was a third distribution of DoBs, e.g. nuclear with 60% (information of second order), and was combined with the result of the first step.

The attribute *CBH* (current best hypothesis) of contacts and clusters is that part of *Prior Hypothesis* with the highest DoB. Contacts are linked to that node of the taxonomy tree which corresponds to their *CBH*.

---

[3] For internal reasons, evidences and clusters are linked to superimposed items; the user should not worry about these items.

The item **OTHER DISJUNCTIONS** comes into play because the evaluation of evidence will not necessarily lead to contact *CBH*s which are nodes of the taxonomy tree. When they are not, the contacts are displayed as linked to the item **OTHER DISJUNCTIONS** of the tree. An example is given in Fig. 9. Two evidences are pooled. The first leads to the *CBH 'a Soviet submarine but not Juliett'*, the second to the *CBH 'Diesel'*. The combination of these evidences leads to *'Diesel but not Juliett'* which is not a node of the tree, i.e. the contact belongs to a disjunction and not a node.

If the user wishes to have this disjunction displayed, he can move the cursor onto the contacts name, select send message from the pop-up menu as it appears and then move the cursor to CBH Expand of the message type menu. As the result, the contact will be linked to all elements of its *CBH* disjunction (Fig. 10). Moving the cursor on CBH Shrink will reinstall the original display.

The item **EVIDENCE** and the taxonomy tree were already discussed in Sect. 2. Rules and their hierarchy will be discussed in Subsect. 4.4.

3.2. USING EVA: EXAMPLES OF HOW IT WORKS

In the EXPLORERs KEE-mode the model can be loaded by typing the name of the knowledge base.

An example scenario has been selected in order to demonstrate the performance of the model. The scenario consists of 9 evidences (Fig. 11) belonging to 7 contact identifiers (**EVIDENCE.2** and **EVIDENCE.4** were identified by the user to belong to **CONTACT.1**). Figure 12 shows the positions of these evidences; the number within the position circles indicates the identification number of the related contact; time and evaluation (as derived by the system) is given as well. The figure was drawn externally because the EXPLORER's plot facility is not yet available for EVA.

Cluster pruning resulted in the creation of 36 clusters (out of 127 theoretical ones). This relatively high number reflects the fact that the information provided by the user is rather poor (6 evidences without a specified attribute) and that the contacts are located closely together.

Insertion of information. The first steps of the scenario's creation will serve to show how information can be inserted by the user via a dialogue with the system.

Figure 13 is a snapshot of the screen after the first three evidences were provided. The whole dialogue can be seen in the typescript window; uppercase letters indicate the system's prompts, lowercase the user's answers.

The dialogue is initiated by pointing a cursor to the New Evidence Creation item which is displayed at the top of the screen. At first the user is asked if he can relate his information to an already existing contact. If his answer is no, the system creates a new contact and prints its name (e.g. CONTACT.1); otherwise it asks the user for the name of the contact to which the information relates.

Next the user is asked for a location. The *Location* attribute has three components: time and x and y coordinates, and they must be given in the form 'txy'. The next prompt is for an attribute.

Any one of the slot names shown in Fig. 2 can be given as an attribute; in addition nil can be answered if no information about attributes is available. If the user's answer is not nil, the system will prompt him for the attributes value; e.g. if the attribute is *Signal*, which means an intercepted radiation, the value can be '*A*', '*B*' or '*C*', each of which stands for a certain peculiarity of the radiation.

When the user has entered all this requested information, the system starts its evaluation and updates contacts and clusters if applicable.

Some first results. After the first three evidences were inserted, a quick look at the knowledge-base graph (Output window of Fig. 13) will provide some first results. It can be seen that CONTACT.1's *CBH* was derived as '*Juliett*' while the evaluation of CONTACT.2 is still completely unknown. The two contacts have no cluster in common, i.e. there have been some properties found by the system which inhibit the conclusion that they may belong to one target. A more detailed inspection would show that speed is the reason for this inconsistency: a constant speed of less than 5 kn could be derived from the two evidences for CONTACT.1, but it would require a speed of more than 5 kn to get from the location of CONTACT.1 to that of CONTACT.2.

Figure 14 is a snapshot of CONTACT.1's attributes at present. Note that values for *CBH*, *Course*, *Speed*, *Inconsistencies* and *Exposure* were derived. [4]

In order to give better understanding of clusters let us step forward to EVIDENCE.5 (see Fig. 11). A part of the resulting knowledge-base graph is depicted in Fig. 15. Now there is a cluster with more than one contact: CONTACT.2 and CONTACT.3 are combined in CL.6. Figure 16 gives all the attribute's values which have been derived for this cluster. Note the value of *Consistent Transition Speed*; this says that it is possible to transit between the member contacts if a target's speed TS is within an interval $5 < TS \leq 10$ kn.

[4] For internal reasons the attribute *Evaluation* does not belong to the contact but to its most recent evidence. Because the latter has a nil attribute, *Evaluation* has a value of complete vagueness.

Derivation of the subsurface picture (SSP). To make things more interesting let us assume that all evidences of the example scenario are now availale to the system. The system has now 36 clusters created; from that confusing number it has to select those candidates which are most likely to be real target tracks.

This performance is invoked by the user pointing the cursor to the Track Assessor's SSP Proposal item which is displayed at the top of the screen. By this action another dialogue via the Typescript window is initiated.

The model displays those member contacts of a cluster which it assumes to be most likely a real track and asks the user if he agrees. If he does, the model takes another likely cluster, etc., until every contact has been displayed once (and of course only once: the SSP has to be unambiguous) in an accepted cluster. Figure 17 shows this dialogue and the resulting SSP plot for the example scenario. The user should now take a look at the *CBH* of these clusters to complete the picture.

If the user rejects a cluster proposal, the system will look for the next most likely ones which also contain the displayed contacts (see Fig. 18). This facility is useful when the user has some expertise not yet implemented in the model: for example, rules have not yet been implemented dealing with the CPA (closest point of approach) but the user may wish to reject a cluster because it would lead to an unacceptable CPA to another track. Another application of this facility is the possibility of introducing false alarms before the model has detected them. The user has merely to reject all clusters which contain the doubtful contact (e.g. `CONTACT.4` in Fig. 19).

Rejection of clusters does not change the knowledge base, and so the user is free to use several SSP proposals by rejecting clusters as he likes, without introducing false information into the knowledge base.

Conflicting evidence. In order to show the impact of conflicting evidence, let the scenario be extended by a tenth evidence (Fig. 20).

This evidence states that `CONTACT.7` (which was previously evaluated to be a *'Juliett with DoB 80%'*) is *'not a Juliett with DoB 99%'*. The Track Assessor will now propose `CONTACT.7` as a separate target (Fig. 21).

More interesting is the inspection of *CBH* and *Prior Hypothesis* of this contact (Fig. 22). *'Juliett'* has reduced its DoB to 1% while there is now a DoB of 20% for *'Complement of Juliett in TH'*. The cumulated DoBs complement to 100% is committed to *'Conflict'*. The relatively high value of the latter (79%) could be an indication that there is something wrong with that contact and it needs some inquiry.

Dempster/Shafer inquiry. This is another dialogue which is invoked and executed similarly to those previously explained. The trigger is the item Dempster/Shafer Inquiry which is displayed next to the other triggers at the top of the screen. The theory for the Dempster/Shafer Inquiry may be found in Shafer [1].

The dialogue asks the system how strongly it relies, for example, on its *CBH* for a specified candidate (contact or cluster); but the user can also try any other hypothesis as well and the system will answer how strongly this hypothesis holds for the candidate.

The inquiry results in a pair of numbers. The first one is called support and is calculated on the basis of all DoBs in favour of the tried hypothesis; e.g. if the hypothesis Diesel is tried and there is some DoB that the candidate is Juliett, then this DoB is also considered to be in favour, because Juliett belongs to Diesel.

The second number is called plausibility. It is calculated on the base of the complement of the cumulated DoBs against the tried hypothesis, i.e. the DoB that there is nothing against the hypothesis.

Some examples are displayed in Fig. 23; they are based on the situation after the conflicting **EVIDENCE.10** was added. The basic properties of the candidate **CONTACT.7** are shown in Fig. 22. At first the previous *CBH* Juliett is tried; support as well as plausibility are quite low (5%). Next, one of the *'not Juliett'*s is tried: support of 0% (because there is no DoB in favour), but a rather high plausibility of 95% (because only the low DoB in Juliett is against the tried hypothesis Echo). The big difference between support and plausibility indicates a high amount of vagueness; there is nothing for and almost nothing against it. The following trials give an impression of what kind of hypotheses can be formulated.

3.3. AUXILIARY FUNCTIONS

These functions have access to the knowledge base but are called from outside the KEE system; the only reason for putting them there was to avoid having too many items (triggers) on the screen. They have to be invoked by the user on the LISP Listener window via the normal LISP function call.

- The Show Evidences function displays all evidences with their attributes and values. An example of the function call can be seen in Fig. 11.
- The Re-assess function allows a re-assessment of a contact, if something in the knowledge base has changed; e.g. there were new rules introduced or a previously supplied evidence was found to be erroneous. An example of the latter is given in Fig. 24: **EVIDENCE.9** of our scenario was found to be wrong; via the editing facility the value of attribute signal was corrected from B to A (see typescript window). Because this evidence concerned **CONTACT.7**, the

latter had to be re-assessed. The syntax for invoking the re-assessment can be seen in the LISP Listener window. The result is that `CONTACT.7` is now under other disjunctions.

- The Reset-all function deletes all information and its derivation. It starts up the model from new. A function call is shown in the LISP Listener window of Fig. 3.

4. Methodology

4.1. CLUSTER PRUNING

To avoid combinatorial explosion in creating clusters, all new or updated contacts have to pass through filters. These filters compare certain properties of the contact with those of already existing clusters. Only if a new contact matches a cluster are they combined to an additional cluster, and only if an updated contact still fits to a cluster that it belonged to previously is this cluster updated as well; otherwise it is deleted. The filters used in the model are the history filter, taxonomy filter and speed filter.

History filter: Whenever a contact was found to be not compatible with another contact, this fact is stored as an attribute of both inconsistencies, termed. The inconsistencies could have been detected by the other two filters or they could have been directly supplied by the user via new evidence. Clusters inherit the inconsistencies of their member contacts.

The history filter uses the inconsistencies attributes of clusters and contacts to prevent cluster creation if a combination which was not possible before becomes possible after a contact's updating. The latter can happen if two hypothetical tracks approach each other.

With user-supplied inconsistencies, the filter is used for deleting clusters which have now become inconsistent (updated contact) or for preventing the creation of inconsistent clusters (new contact).

Taxonomy filter: This filter prevents the combination of contacts which are not compatible with respect to their *CBH*s. The filter not only regards the candidate *CBH*s (information of first order) but looks also for a third *CBH* which is derived from the hypothetical transition speed between the candidates (information of second order). All three *CBH*s have to be mutually consistent. Consistency in this context means that all *CBH*s are on the same branch of the taxomony tree.

Speed filter: Usually the maximum speed of hypothetical targets is used to decide whether or not two contacts could belong to the same track. This approach makes sure that the set of clusters contains those clusters which are *entire* tracks of detected targets. The disadvantage of this approach is that it is too coarse for efficient cluster pruning.

To compensate for the coarseness the model restricts cluster creation by only allowing clusters for which the mutual transition speeds between their constituent contacts are within a certain speed interval. At present the consistent transition speed attribute of a cluster is graduated in 5 kn steps.

Of course, an entire track can now be achieved only if the target remains within its initial speed interval, as otherwise the track is decomposed in more than one cluster, with each of these showing the instantaneous target speed as its consistent transition speed. This disadvantage is not of great weight because the model's SSP proposal will display this decomposition such that the user will easily recognize the entire track just by the cluster's alignment.

The speed filter considers (1) the speed attribute of the contact, (2) the consistent transition speed attribute of the cluster, (3) the transition speed between the contact and the most recently added (or updated) member contact of the cluster. All three speeds have to be mutually consistent, i.e. they have to be in the same interval.

4.2. SSP PROPOSAL

Even after filtering has reduced the number of clusters considerably, the need to distinguish real tracks from garbage still remains and is a challenging task.

An approach could be to regard the time-ordered sequence of contacts (or their evidences) as a dynamically developing situation and to apply target models to find out if any kind of target will match the situation (and to what degree). One problem with that approach is that targets are very difficult to model due to their behaviour. [5] Another problem is the low data rate within a cluster; in consequence a lot of a target's inherent dynamics are unknown and several target models could match a cluster, even at one which reflects which is not real track.

Another approach could be to assign a likelihood of existence to any cluster and to discard all clusters with likelihoods below a certain threshold. This likelihood can be derived from two constituents every time a contact is updated or newly added to the cluster. The first constituent is the likelihood of existence of the new evidence, the second is the likelihood that a situation, reflected by the original cluster, will change to another situation, reflected by the updated cluster. The only source for both likelihoods is the judgement of experts. While the existence likelihood will be obtained quite easily for a broad spectrum of expected evidences and under various conditions, the situation-change likelihood will be almost impossible to obtain for wartime situations.

[5] This behaviour may be predictable in peacetime. During war preventive actions and reactions to search efforts will render it almost unpredictable. At least targets will then avoid any linearities which use a variety of analysis aids such as Kalman filters.

For these reasons the model uses a solution which is independent of target model and expert judgements. The basic idea is that clusters representing real tracks can be identified using heuristic rules. Two such rules could be derived looking at the sets of clusters generated for different scenarios:

Rule 1: Clusters with a high cardinality are more likely to correspond to real tracks than clusters with few members. It can be assumed that the previous filtering has ruled out many high cardinality clusters which are not real tracks so that we can expect the real tracks to lie among the larger of the remaining clusters.

Rule 2: A cluster is valid if and only if all contacts within this cluster do not lie within another cluster which is not a subset of this valid cluster.

Using these rules each cluster is checked against all the others.

The greater the number of these other clusters that are subsets of the candidate cluster, the stronger is the indication that the candidate is valid. The greater the number of clusters are found to share contacts with the candidate but are not subsets of it the stronger is the indication that the candidate is false. A rank is obtained for each cluster by combining both indications.

The model now takes the cluster with the highest rank and proposes it to the user as a track. If the user accepts, the model looks for the cluster with the highest rank among those that remain and do not share any contact with the accepted one; the model then proposes it to the user. This procedure is iterated. After all contacts are covered by accepted clusters, the model displays this selection of clusters as its SSP proposal.

4.3. TREATMENT OF FALSE ALARMS

False alarm is an item of the taxonomy tree and so it is treated like other evaluations of contacts and clusters, i.e. it is accessible to evaluation rules and evidence pooling.

In consequence, the *CBH* of a contact or a cluster may become a false alarm. If this happens to a cluster, the model will delete it; if it happens to a contact, all clusters which contain this contact are deleted and the contact is linked to the false alarm node of the taxonomy tree for eventual backtracking. This deletion and linking represents an additional cluster pruning.

The SSP proposal of the model does not consider the possibility of false alarms, i.e. no degenerated tactical pictures are *a priori* proposed. This results in a considerable reduction in the number of possible tactical pictures. If the user wishes to take account of false alarms, he can do so through engaging in dialogue with the model's Track Assessor item, from which he rejects cluster proposals which seem doubtful to him or contain doubtful contacts (Fig. 19); as a decision aid he can look at the cluster's or contact's *Prior Hypothesis* attribute.

4.4. RULES

In the present state of the EVA prototype there are two major classes of rules:

(a) evaluation rules which are applied each time a new evidence appears,

(b) validation rules which are applied by the Track Assessor item in order to distinguish between valid and false clusters.

At present, evaluation rules can derive high-level attributes of contacts from lower level ones (e.g. the exposed antenna from an intercepted signal) in combination with evaluations from speed or exposure. Up to now, the only validation rule deals with transitting targets and states that a cluster is unlikely to be a real track if the time sequence of its member contacts indicates heavy manoeuvering. (Fig. 25). [6]

Rule application follows the backward chaining strategy: a primary goal is posed. In the case of the class of evaluation rules this goal is the answer to the major question: 'The evaluation of (evidence) is *what*?' At first the model looks to see if there is already an explicit answer; if not, it takes one of the rules whose conclusion provides such an answer and poses the verification of the rules premise as a subgoal. Then the model looks to see if there is already an explicit verification; if not, it takes a rule whose conclusion matches with the premise of the previous one and poses the verification of the new premise on the subgoal stack, etc. The model goes backward until finally it finds a verification; all conclusions of that chain of rules are now asserted and an evaluation is derived. If no verification can be found within a chain, the model goes forward again to the primary goal, takes the next rule with an appropriate conclusion and chains backward again, etc.

4.5. DERIVATION OF DOB BY EVIDENCE POOLING

A quotation from S.D. Poisson [2] once more characterizes the subjective nature of DoB:

> *Ainsi, un évènement aura, par sa nature, une chance plus ou moins grande, connue ou inconnue ; et sa probabilité sera relative à nos connaissances en ce qui le concerne.*

DoB has to be understood here as the notion 'probabilité' in the above quotation.

[6] The rules in the present prototype will serve only as templates for further extensions of the model.

The syntax of the *Evaluation* attribute of contacts and clusters comprises the event ('évènement': here the presence of a target, specified by its evaluation) and the DoB (in the notation of a percentage); see Fig. 4 for an example. The *Evaluation* attribute may consist of an arbitrarily extensive list of such event/DoB pairs, if the information does not allow focusing of the DoB on a unique event.

Such lists are the elements of evidence pooling. They are supplied either by the user directly or by the system. When supplied by the user directly it is as an evidence's attribute (see evidence No. 5 in Fig. 11). When derived by the system, it is through:

(a) applying evaluation rules on other types of evidence's attributes (Fig. 25), or

(b) using second-order information, or

(c) transferring the contacts evaluation to clusters.

These lists are combined with an already existing list, the *Prior Hypothesis* attribute of the contact or cluster in question, to form an updated *Prior Hypothesis*. The method used is Dempster's rule of combination as presented in [1].

In the terminology of [1], the DoB part of an *Evaluation* attribute is expressed as a basic probability assignment; the updated *Prior Hypothesis* attribute is called the orthogonal sum of the pooled evidences.

When using this method, the user has to take some care. Firstly he has to assure that the supplied evidences are independent. Secondly he has to define the case thoroughly (according to Example 3 in Subsect. 2.2.) before distributing his DoB. For example, if he decides to use the conventional Bayesian case, he has to assign the appropriate basic probability explicity to the complement of his hypothesis as well. Or if he wants to specify a refinement, he has to keep in mind that a refinement hypothesis implies the coarse one—i.e. if his DoB in Diesel is 90% and he believes that the Diesel may be a Juliett with DoB 60%, he has to assign to Diesel a basic probability of only 30%, because 60% are already committed to it via Juliett (which is a Diesel).

The following part is interesting only for a reader who is familiar with Shafer's theory of evidence.

The model has implemented Shafer's procedure in a slightly modified manner. The orthogonal sum is not normalized after each step. Alternatively the Prior Hypothesis attribute is extended by an element conflict where all occurring assignments to conflicting combinations are cumulated (Fig. 26). Only when the user invokes Dempster/Shafer Inquiry is a transient normalization carried out for the calculation of support and plausibility.
The reason for transient normalization is twofold. Firstly the model can now be extended to use conflict (e.g. in rules for defining false alarms) without changing an algorithm. Secondly normalization conceals the existence of conflict. Prade [3] and Zadeh [4] have pointed out cases where this can lead to strange results; thus, treatments of conflict other than merely normalization can be expected as studies go on and the model evolves.

5. Summary

The expert system prototype EVA derives contact and target evaluations and target tracks (i.e. subsurface pictures) from user-supplied evidences. These evidences may be unreliable, imprecise, incomplete and even mutually inconsistent.

The evaluations are presented by EVA as the current best hypotheses in terms of nodes of the taxonomy tree or of disjunctions of the tree's top. The system pools all evidences in order to raise the current best hypotheses to as high a level as possible and to increase the degree of belief in this hypothesis.

The tracks are presented as clusters of contacts which are most likely to belong to an individual target.

References

[1] SHAFER, G. A Mathematical Theory of Evidence. Princeton N.J., Princeton University Press, 1976.

[2] POISSON, S.D. Recherches sur la probabilité des jugements. 1837.

[3] PRADE, H. A computational approach to approximate and plausible reasoning with applications to Expert Systems. *In*: IEEE Transactions on Pattern Analysis and Machine Intelligence, 1985.

[4] ZADEH, L.A. Review of a mathematical theory of evidence by G. Shafer. *In*: *AI Magazine*, Fall, 1984.

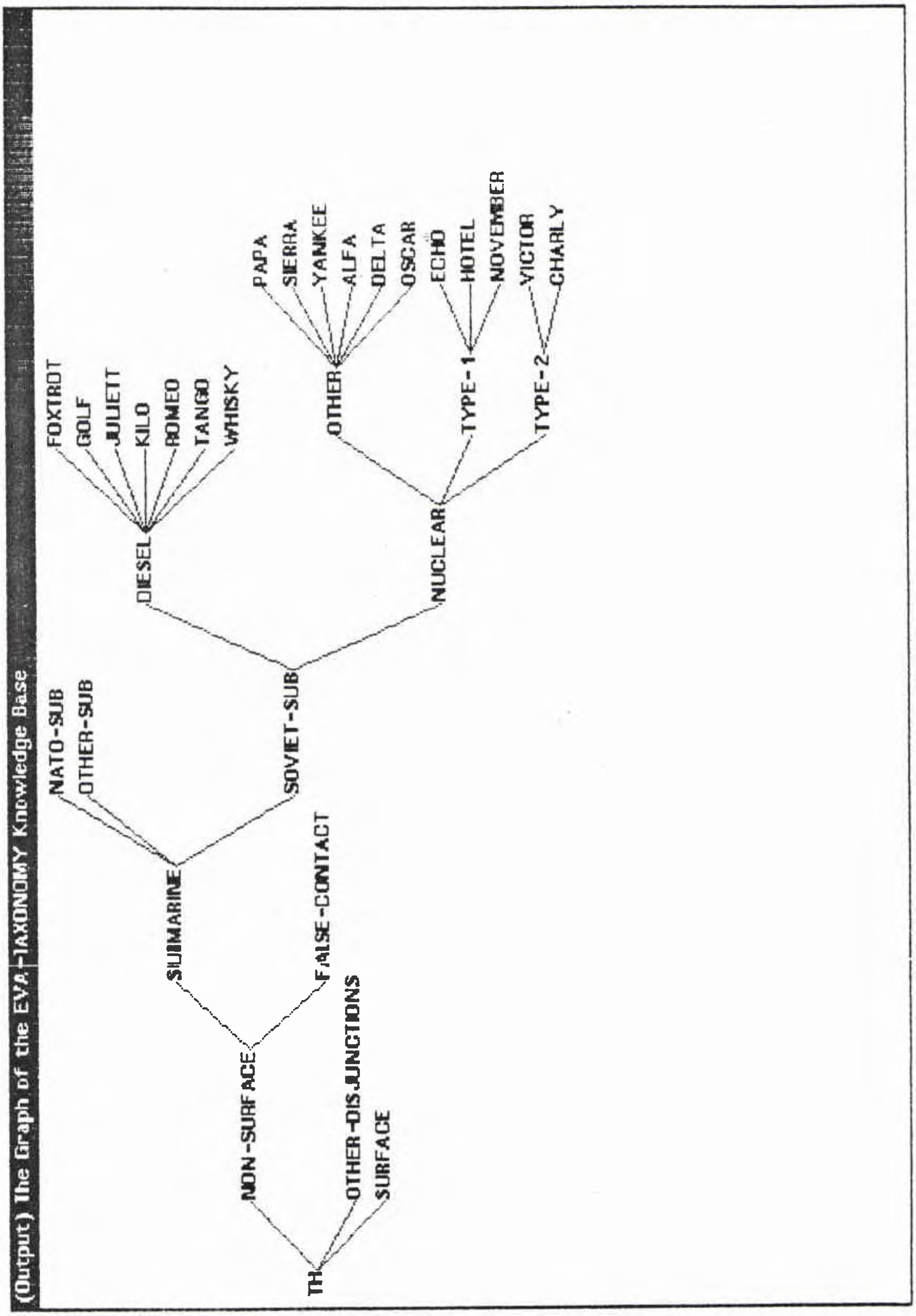

Fig. 1. Taxonomy tree.

```
(Output) The EVIDENCE.1 Unit in EVA Knowledge Base
Unit: EVIDENCE.1 in knowledge base EVA
Created by  WERNER on 16-Oct-1986 8:12:29
Modified by  WERNER on 16-Oct-1986 13:27:06
Member Of: EVIDENCES

Own slot: CONTACT from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: CONTACT.1

Own slot: COURSE from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: EVALUATION from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: EXPOSURE from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: INCONSISTENCIES from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: LOCATION from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: (0 0 0)

Own slot: SIGNAL from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: B

Own slot: SPEED from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: STATE from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: SUB.TASK from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: TIME.OF.CRS.CHG from EVIDENCE.1
   Inheritance: OVERRIDE.VALUES
   Values: Unknown
```

Fig. 2. Example of attributes of evidence.

Report no. changed (Mar 2006): SR-123-UU

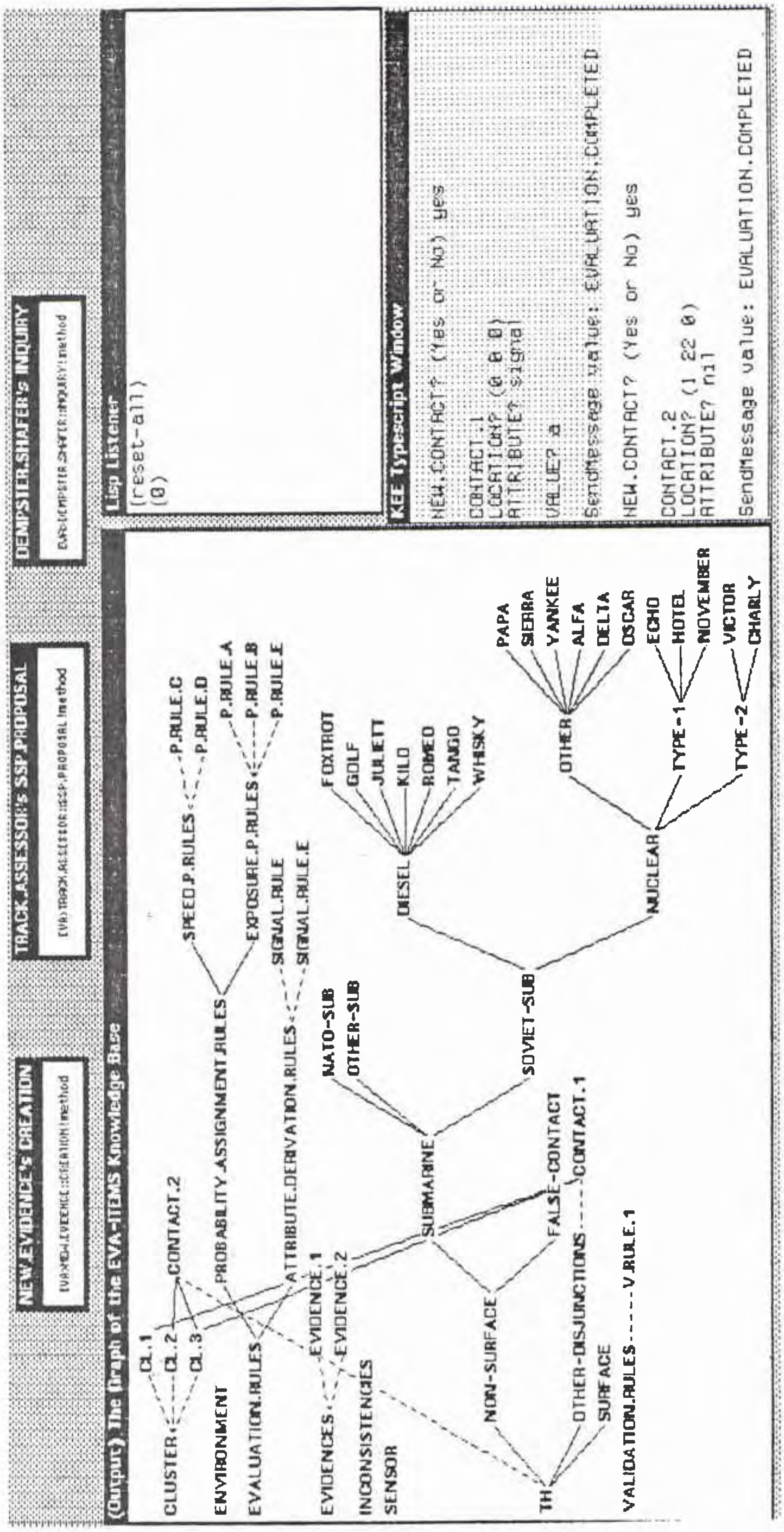

Fig. 3. Model display.

```
(Output) The CONTACT.1 Unit in EVA Knowledge Base
Unit: CONTACT.1 in knowledge base EVA
Created by  WERNER on 17-Oct-1986 10:54:32
Modified by  WERNER on 17-Oct-1986 14:16:41
  Member Of: (ENTITIES in kb GENERICUNITS)

Own slot: CBH from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: (HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA
            OSCAR SIERRA FOXTROT TANGO KILO WHISKY ROMEO GOLF)

Own slot: EVALUATION from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: (((COMPLEMENT JULIETT SOVIET-SUB) 95))

Own slot: EVIDENCES from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: (EVIDENCE.1 EVIDENCE.1)

Own slot: EXPOSURE from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: SNOOP-TRAY

Own slot: LOCATION from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: (0 0 0)

Own slot: PRIOR.HYPOTHESIS from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: (((CONFLICT) 0)
            ((SURFACE FALSE-CONTACT OTHER-SUB NATO-SUB HOTEL ECHO
              NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA
              FOXTROT TANGO KILO JULIETT WHISKY ROMEO GOLF) 5)
            ((HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA
              OSCAR SIERRA FOXTROT TANGO KILO WHISKY ROMEO GOLF) 95))

Own slot: SIGNAL from CONTACT.1
   Inheritance: OVERRIDE.VALUES
   Values: A
```

Fig. 4. Example of derived values for a contact.

```
(Output) The CL.3 Unit in EVA Knowledge Base
 Unit: CL.3 in knowledge base EVA
Created by   WERNER on 17-Oct-1986 10:55:39
Modified by   WERNER on 17-Oct-1986 14:24:40
  Superclasses: (ENTITIES in kb GENERICUNITS)
  Subclasses: CONTACT.1, CONTACT.2
  Member Of: CLUSTER, (CLASSES in kb GENERICUNITS)

Own slot: CBH from CL.3
   Inheritance:  OVERRIDE.VALUES
   Values: (HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA
             OSCAR SIERRA)

Own slot: CONSISTENT.TRANSITION.SPEED from CL.3
   Inheritance:  OVERRIDE.VALUES
   ValueClass: (NUMBER in kb KEEDATATYPES)
   Cardinality.Min: 1
   Cardinality.Max: 1
   Comment: "interval: value-5, value"
   Values: 25

Own slot: COURSE from CL.3
   Inheritance:  OVERRIDE.VALUES
   Values: 0.0

Own slot: COURSE.CHANGE from CL.3
   Inheritance:  OVERRIDE.VALUES
   Values: NIL

Own slot: DECOMPOSITION.COMPLETE from CLASSES
   Inheritance:  UNION
   ValueClass: (LIST.OF (CLASSES in kb GENERICUNITS))
   Comment: "
   A complete decomposition is a list of
   subclasses of this class which jointly
   subsume all the members of this class.
   More than one complete decomposition
   may be specified.
"
   Values: Unknown

Own slot: DECOMPOSITION.DISJOINT from CLASSES
   Inheritance:  UNION
   ValueClass: (LIST.OF (CLASSES in kb GENERICUNITS))
   Comment: "
   A disjoint decomposition is a list of
   subclasses of this class which share no
   members.  More than one disjoint
   decomposition may be specified.
```

Fig. 5. Example of a cluster, part 1.

```
(Output) The CL.3 Unit in EVA Knowledge Base
Own slot: INCONSISTENCIES from CLUSTER
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: MEMBERS.DATATYPE from CLASSES
   Inheritance: OVERRIDE
   ValueClass: (DATATYPE in kb KEEDATATYPES)
   Values: (UNIT in kb KEEDATATYPES)

Own slot: MEMBERSHIP from CLASSES
   Inheritance: METHOD
   ValueClass: (METHOD in kb KEEDATATYPES)
   Cardinality.Max: 1
   Cardinality.Min: 1
   Values: MEMBER-DISJOINTP!METHOD

Own slot: PRIOR.HYPOTHESIS from CL.3
   Inheritance: OVERRIDE.VALUES
   ValueClass: (LIST in kb KEEDATATYPES)
   Cardinality.Min: 1
   Cardinality.Max: 1
   Values: (((CONFLICT) 0)
           ((HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA
             OSCAR SIERRA FOXTROT TANGO KILO WHISKY ROMEO GOLF) 38)
           ((SURFACE FALSE-CONTACT OTHER-SUB NATO-SUB HOTEL ECHO NOVEMBER
             CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA FOXTROT
             TANGO KILO JULIETT WHISKY ROMEO GOLF) 2)
           ((HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA
             OSCAR SIERRA) 60))

Own slot: SUBCLASSP from CLASSES
   Inheritance: METHOD
   ValueClass: (METHOD in kb KEEDATATYPES)
   Cardinality.Min: 1
   Cardinality.Max: 1
   Values: SUBCLASS-DISJOINTP!METHOD

Own slot: TIME.OF.CRS.CHG from CLUSTER
   Inheritance: OVERRIDE.VALUES
   Values: Unknown

Own slot: VALIDITY from CLUSTER
   Inheritance: OVERRIDE.VALUES
   Values: 1.0
```

Fig. 6. Example of a cluster, part 2.

```
(Output) The ENVIRONMENT Unit in EVA Knowledge Base
 Unit: ENVIRONMENT in knowledge base EVA
Created by   WERNER on 15-Jul-2186 6:58:18
Modified by   WERNER on 17-Oct-1986 14:35:01
  Superclasses: (ENTITIES in kb GENERICUNITS)
  Member Of: (CLASSES in kb GENERICUNITS)

Member slot: LAYER from ENVIRONMENT
    Inheritance:  OVERRIDE.VALUES
    ValueClass: (ONE.OF ZERO WEAK STRONG)
    Values: Unknown

Member slot: LAYER.DEPTH from ENVIRONMENT
    Inheritance:  OVERRIDE.VALUES
    ValueClass: (NUMBER in kb KEEDATATYPES)
    Values: Unknown

Member slot: PRECIPITATION from ENVIRONMENT
    Inheritance:  OVERRIDE.VALUES
    ValueClass: (ONE.OF ZERO DRIZZLE RAIN HEAVY.RAIN)
    Values: Unknown

Member slot: SEASTATE from ENVIRONMENT
    Inheritance:  OVERRIDE.VALUES
    ValueClass: (NUMBER in kb KEEDATATYPES)
    Values: Unknown

Member slot: VISIBILITY from ENVIRONMENT
    Inheritance:  OVERRIDE.VALUES
    ValueClass: (NUMBER in kb KEEDATATYPES)
    Values: Unknown
```

Fig. 7. Environmental attributes.

```
(Output) The SENSOR Unit in EVA Knowledge Base
 Unit: SENSOR in knowledge base EVA
Created by   WERNER on 15-Jul-2186 6:58:18
Modified by   WERNER on 15-Jul-2186 6:59:31
  Superclasses: (ENTITIES in kb GENERICUNITS)
  Member Of: (CLASSES in kb GENERICUNITS)

Member slot: PLATFORM from SENSOR
    Inheritance:  OVERRIDE.VALUES
    Values: Unknown

Member slot: TYPE from SENSOR
    Inheritance:  OVERRIDE.VALUES
    Values: Unknown
```

Fig. 8. Sensor attributes.

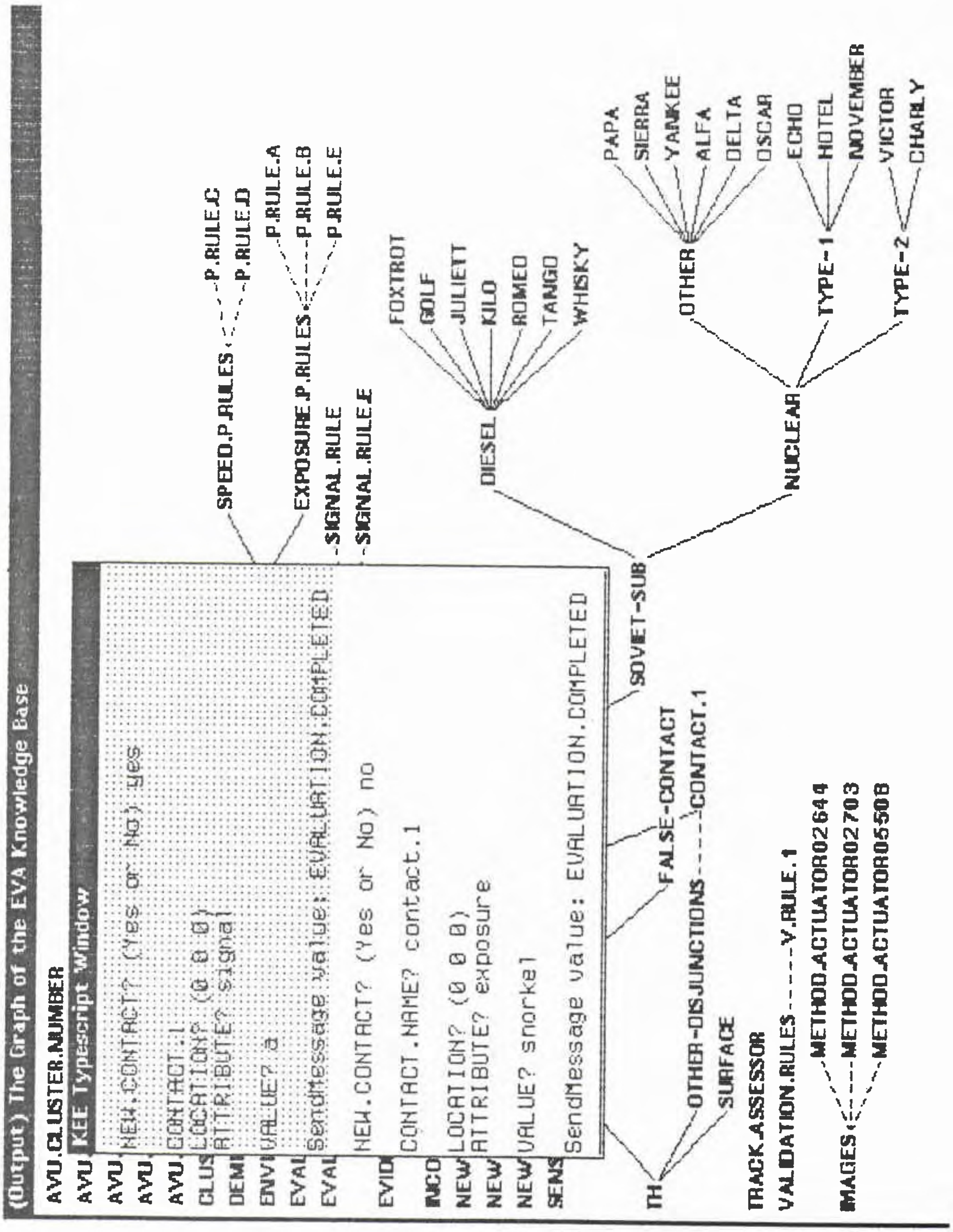

Fig. 9. Example of Other Disjunctions.

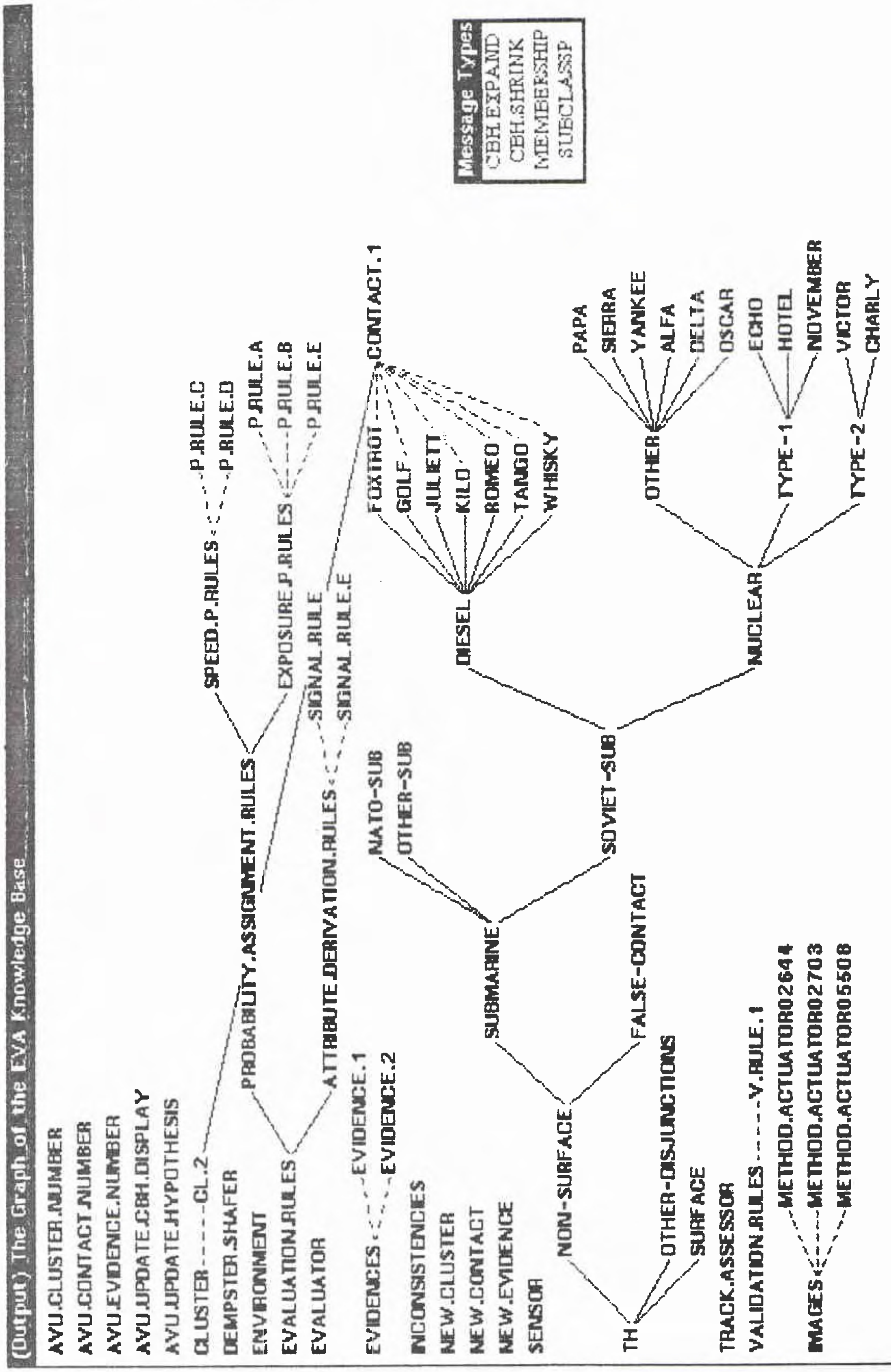

Fig. 10. Expanded view of Other Disjunctions example.

```
Lisp Listener
(show-evidences)
EVIDENCE.1
CONTACT.1 (6 2 13)
SIGNAL (8)
EVIDENCE.2
CONTACT.1 (7 5 12)
EVIDENCE.3
CONTACT.2 (8 5 18)
EVIDENCE.4
CONTACT.1 (8 9 11)
EVIDENCE.5
CONTACT.3 (9 10 14)
EVALUATION (((NUCLEAR 75)))
EVIDENCE.6
CONTACT.4 (10 14 8)
EVIDENCE.7
CONTACT.5 (10 15 10)
EVIDENCE.8
CONTACT.6 (11 20 6)
EVIDENCE.9
CONTACT.7 (12 21 7)
SIGNAL (8)
NIL

(cardinality (unit.children 'cluster 'member))
36
```

Fig. 11. Example scenario: 9 evidences belonging to 7 contact identifiers.

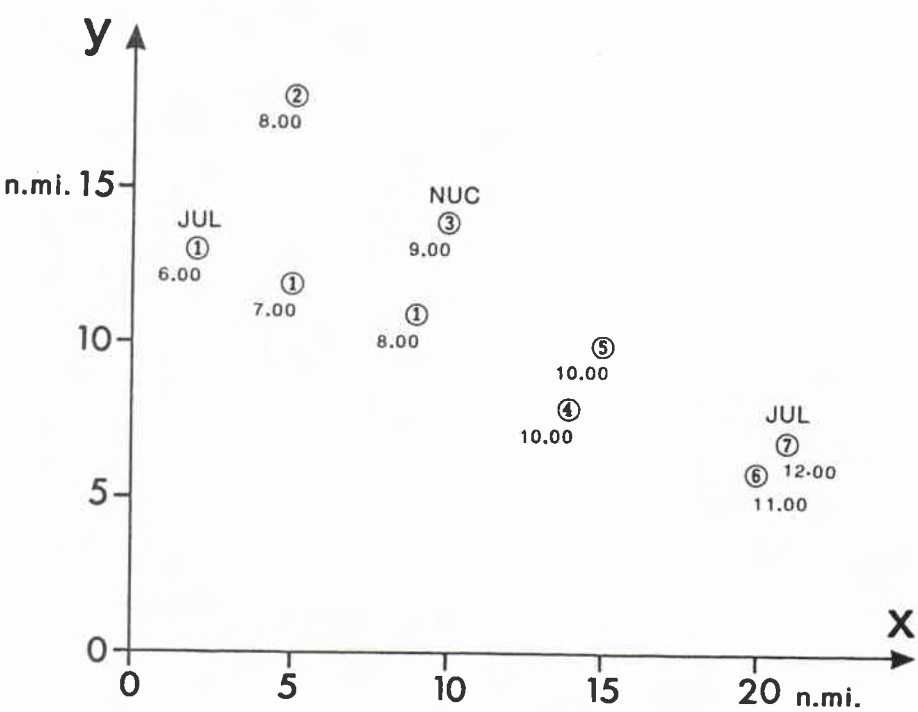

Fig. 12. Example scenario: geographic display of 9 evidences

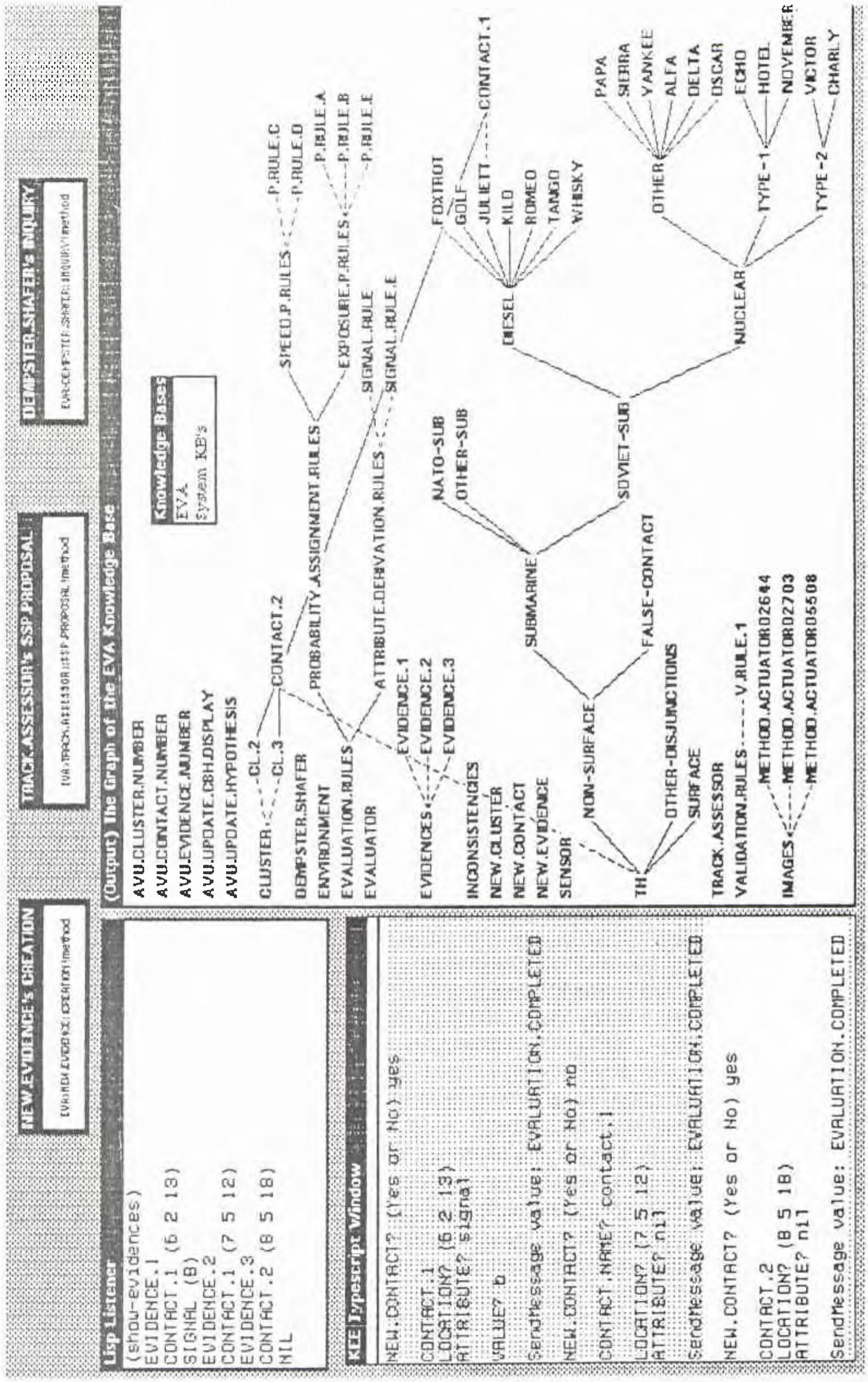

Fig. 13. Example scenario: initial inputs.

```
(Output) The List of the CONTACT.1 Unit in the EVA Knowledge Base
All slots of Unit CONTACT.1 in KB EVA are:
Slot          Role               ValueClass  Value
CBH           OVERRIDE.VALUES    NIL         (JULIETT)
CBH.EXPAND
              METHOD             METHOD      |EVA>TH:CBH.EXPAND!method|
CBH.SHRINK    METHOD             METHOD      |EVA>TH:CBH.SHRINK!method|
COURSE        OVERRIDE.VALUES    NIL         5.96143474
DECOMPOSITION.COMPLETE
              UNION              (LIST.OF (CLASSES in kb GENERICUNITS))
                                             NIL
DECOMPOSITION.DISJOINT
              UNION              (LIST.OF (CLASSES in kb GENERICUNITS))
                                             NIL
EVALUATION
              OVERRIDE.VALUES    NIL         ((TH 100))
EVIDENCES     OVERRIDE.VALUES    NIL         '(EVIDENCE.1 EVIDENCE.1 EVIDENCE.2)
EXPOSURE      OVERRIDE.VALUES    (ONE.OF SNORKEL SNOOP-TRAY SNOOP-SLAB)
                                             SNOOP-SLAB
INCONSISTENCIES
              OVERRIDE.VALUES    NIL         CONTACT.2
LOCATION      OVERRIDE.VALUES    LIST        (7 5 12)
MEMBERS.DATATYPE
              OVERRIDE           DATATYPE    (UNIT in kb KEEDATATYPES)
MEMBERSHIP
              METHOD             METHOD      MEMBER-DISJOINTP!METHOD
PRIOR.HYPOTHESIS
              OVERRIDE.VALUES    LIST        (((CONFLICT) 0) ((JULIETT) 80)
                                              ((SURFACE FALSE-CONTACT OTHER-SUB NATO-SUB HOTEL ECHO NOVEMBER
                                                CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA FOXTROT TANGO
                                                KILO JULIETT WHISKY ROMEO GOLF) 20))
SIGNAL        OVERRIDE.VALUES    (ONE.OF A B C)
                                             B
SPEED         OVERRIDE.VALUES    NUMBER      3.162277661
STATE         OVERRIDE.VALUES    (ONE.OF SURFACED SUBMERGED AT.PERISCOPE SNORKELLING DEEP MISSILE.LAUNCH UPKEEP)
                                             NIL
SUB.TASK      OVERRIDE.VALUES    (ONE.OF PATROL TRANSIT TRAIL SHADOWING)
                                             NIL
```

Fig. 14. Example scenario: snapshot of knowledge base showing contact 1 after 3 evidences.

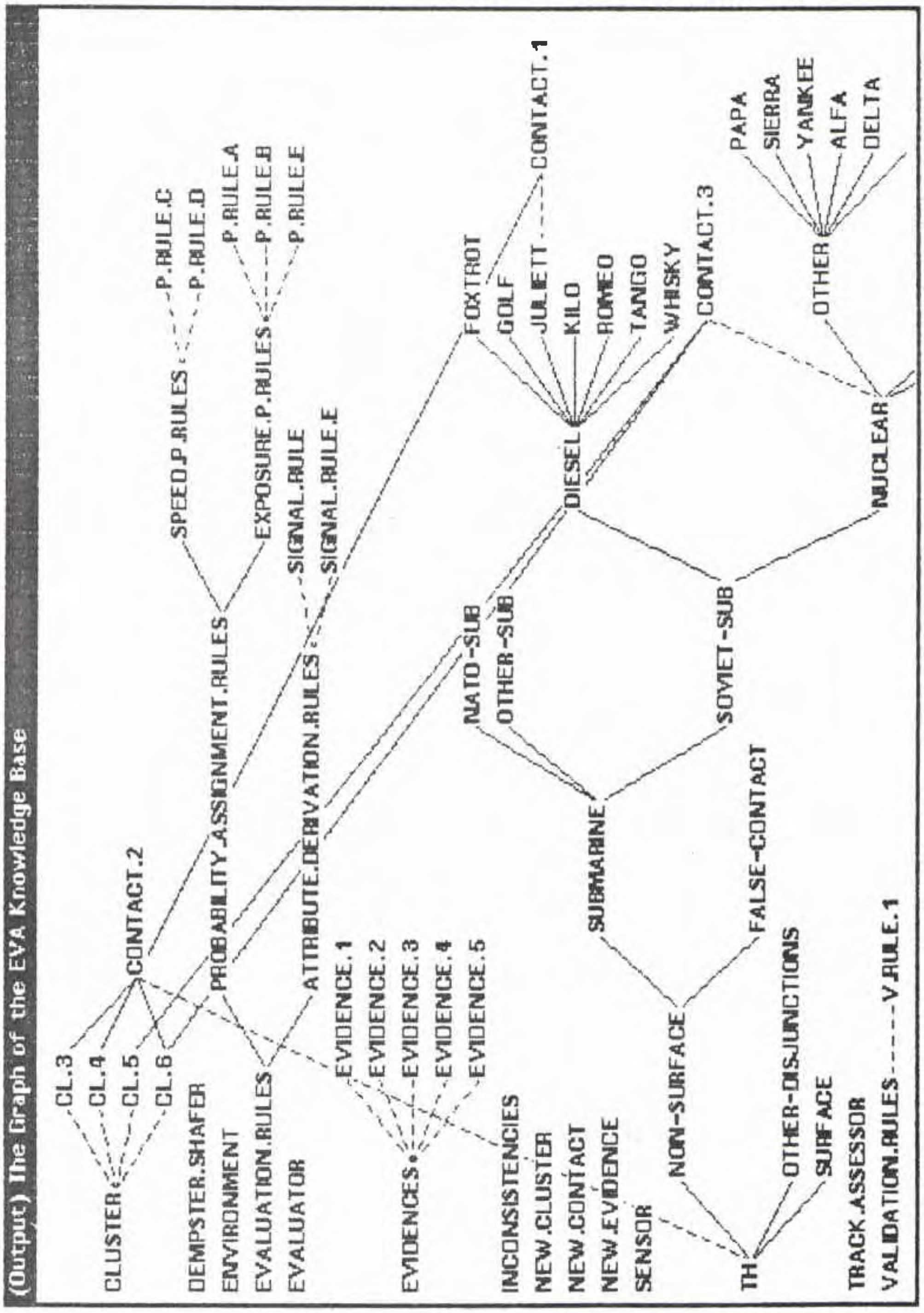

Fig. 15. **Example scenario: snapshot of knowledge base after 5 evidences.**

(Output) The List of the CL.6 Unit in the EVA Knowledge Base

All slots of Unit **CL.6** in KB **EVA** are:

Slot	Role	ValueClass	Value
CBH	OVERRIDE.VALUES	NIL	(HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA)
CONSISTENT.TRANSITION.SPEED			
	OVERRIDE.VALUES	NUMBER	10
COURSE	OVERRIDE.VALUES	NIL	5.60844433
COURSE.CHANGE			
	OVERRIDE.VALUES	NIL	NIL
DECOMPOSITION.COMPLETE			
	UNION	(LIST.OF (CLASSES in kb GENERICUNITS))	NIL
DECOMPOSITION.DISJOINT			
	UNION	(LIST.OF (CLASSES in kb GENERICUNITS))	NIL
INCONSISTENCIES			
	OVERRIDE.VALUES	NIL	CONTACT.1
MEMBERS.DATATYPE			
	OVERRIDE	DATATYPE	(UNIT in kb KEEDATATYPES)
MEMBERSHIP			
	METHOD	METHOD	MEMBER-DISJOINTP!METHOD
PRIOR.HYPOTHESIS			
	OVERRIDE.VALUES	LIST	(((CONFLICT) 0) ((HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA) 75) ((SURFACE FALSE-CONTACT OTHER-SUB NATO-SUB HOTEL ECHO NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA FOXTROT TANGO KILO JULIETT WHISKY ROMEO GOLF) 25))
SUBCLASSP	METHOD	METHOD	SUBCLASS-DISJOINTP!METHOD
TIME.OF.CRS.CHG			
	OVERRIDE.VALUES	NIL	NIL
VALIDITY	OVERRIDE.VALUES	NIL	1.0

Fig. 16. Example scenario: snapshot of Cluster No 6.

```
KEE Typescript Window

CONTACT.3
CONTACT.6
CONTACT.5
CONTACT.2
OK? (Yes or No) yes

CONTACT.7
CONTACT.1
CONTACT.4
OK? (Yes or No) yes

MY.SSP.PROPOSAL

SendMessage value: (CL.20 CL.35)
```

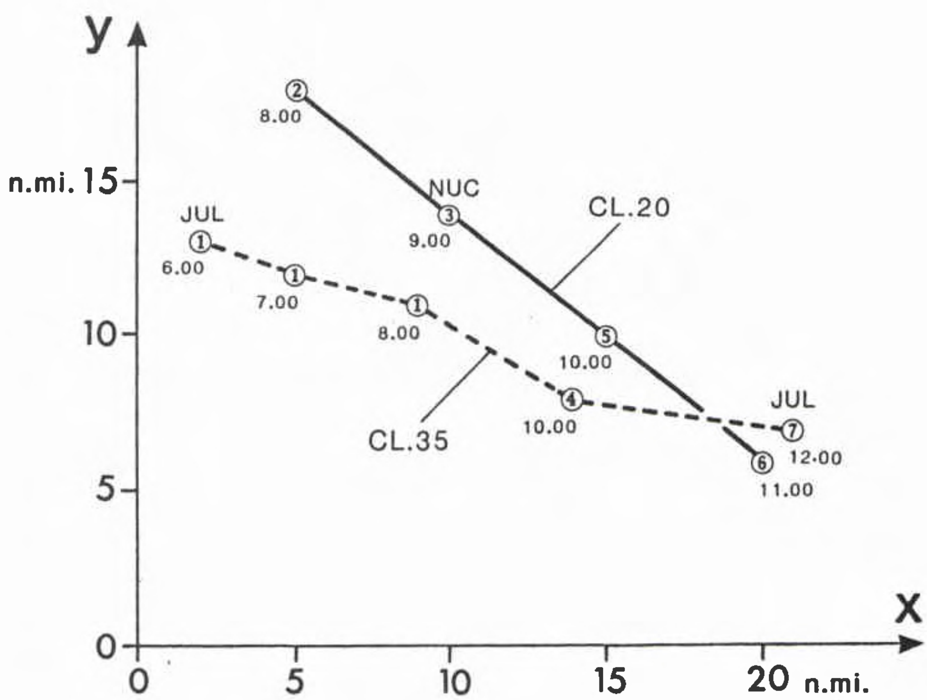

Fig. 17. Example scenario: SSP proposal showing user dialogue (above) and resulting geographic plot.

```
KEE Typescript Window

CONTACT.3
CONTACT.6
CONTACT.5
CONTACT.2
OK? (Yes or No) no

CONTACT.3
CONTACT.6
CONTACT.4
CONTACT.2
OK? (Yes or No) yes

CONTACT.7
CONTACT.1
CONTACT.5
OK? (Yes or No) yes

MY.SSP.PROPOSAL

SendMessage value: (CL.24 CL.33)
```

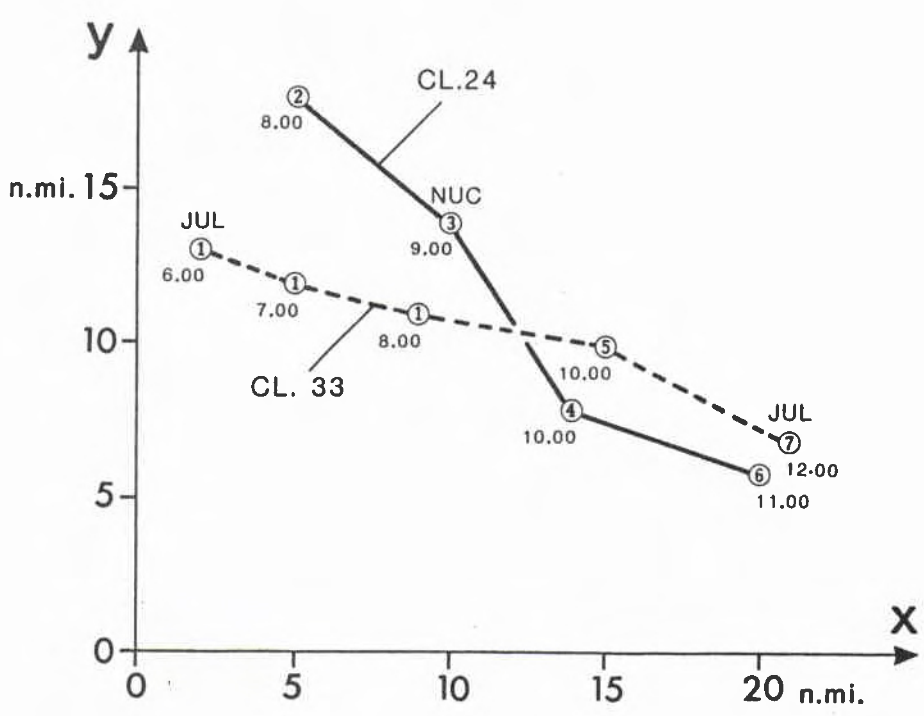

Fig. 18. Example scenario: user influenced SSP proposal showing dialogue (above) and resulting geographic plot.

```
KEE Typescript Window

CONTACT.3
CONTACT.6
CONTACT.5
CONTACT.2
OK? (Yes or No) yes

CONTACT.7
CONTACT.1
CONTACT.4
OK? (Yes or No) no

CONTACT.7
CONTACT.1
OK? (Yes or No) yes

CONTACT.4
OK? (Yes or No) no

MY.SSP.PROPOSAL

SendMessage value: (CL.20 CL.37)
```

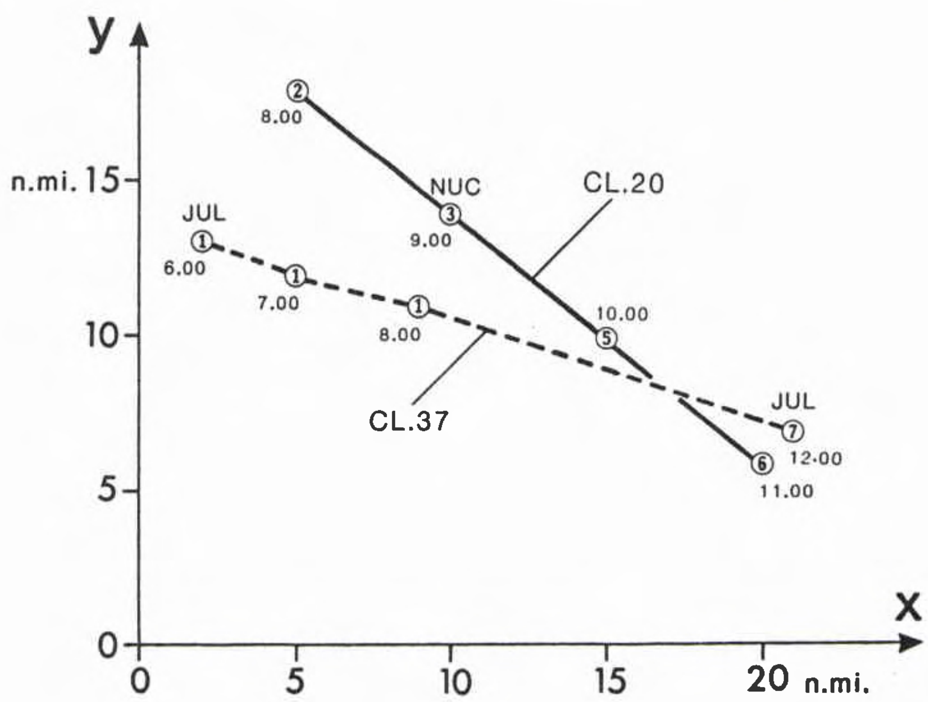

Fig. 19. Example scenario: elimination of a false alarm user with resulting geographic plot.

```
Lisp Listener
(show-evidences)
EVIDENCE.1
CONTACT.1 (6 2 13)
SIGNAL (B)
EVIDENCE.2
CONTACT.1 (7 5 12)
EVIDENCE.3
CONTACT.2 (8 5 18)
EVIDENCE.4
CONTACT.1 (8 9 11)
EVIDENCE.5
CONTACT.3 (9 10 14)
EVALUATION (((NUCLEAR 75)))
EVIDENCE.6
CONTACT.4 (10 14 8)
EVIDENCE.7
CONTACT.5 (10 15 10)
EVIDENCE.8
CONTACT.6 (11 20 6)
EVIDENCE.9
CONTACT.7 (12 21 7)
SIGNAL (B)
EVIDENCE.10
CONTACT.7 (12 21 7)
EVALUATION ((((COMPLEMENT JULIETT TH) 99)))
NIL
```

Fig. 20. Extended example scenario.

```
KEE Typescript Window

CONTACT.3
CONTACT.6
CONTACT.5
CONTACT.2
OK? (Yes or No) yes

CONTACT.1
CONTACT.4
OK? (Yes or No) yes

CONTACT.7
OK? (Yes or No) yes

MY.SSP.PROPOSAL

SendMessage value: (CL.20 CL.10 CL.39)
```

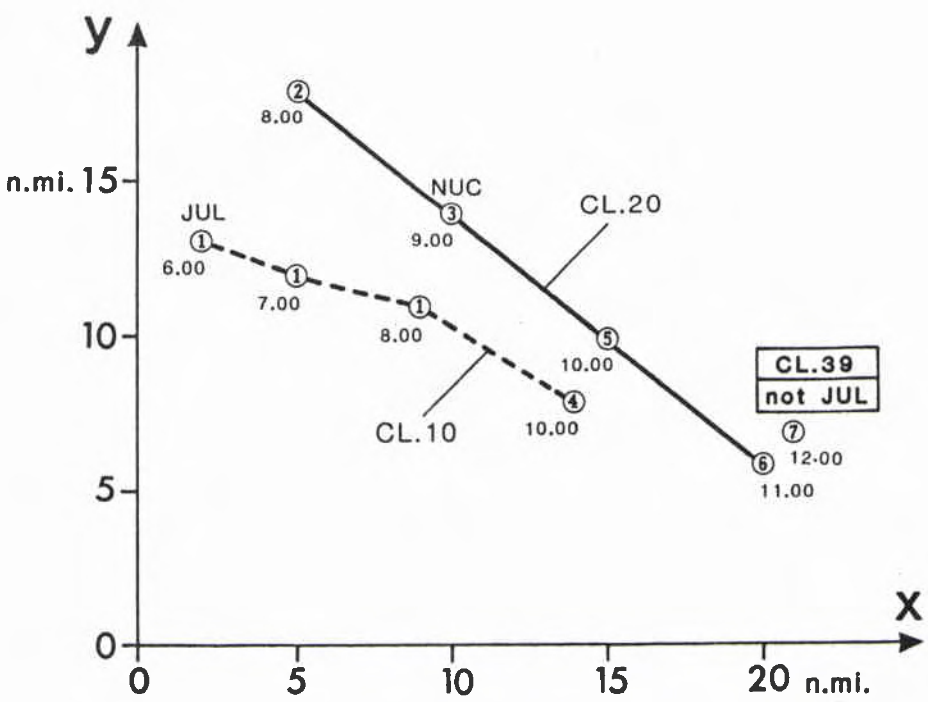

Fig. 21. Extended example scenario: SSP proposal showing dialogue (above) and resulting geographic plot.

Report no. changed (Mar 2006): SR-123-UU


```
(Output) The CONTACT.7 Unit in EVA Knowledge Base
Own slot: PRIOR.HYPOTHESIS from CONTACT.7
  Inheritance: OVERRIDE.VALUES
  ValueClass: (LIST in kb KEEDATATYPES)
  Avunits: AVU.UPDATE.CBH.DISPLAY
  Cardinality.Min: 1
  Cardinality.Max: 1
  Values: (((CONFLICT) 79) ((JULIETT) 1)
          ((SURFACE FALSE-CONTACT OTHER-SUB NATO-SUB HOTEL ECHO
            NOVEMBER CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA
            FOXTROT TANGO KILO WHISKY ROMEO GOLF) 20))

Own slot: CBH from CONTACT.7
  Inheritance: OVERRIDE.VALUES
  Values: (SURFACE FALSE-CONTACT OTHER-SUB NATO-SUB HOTEL ECHO NOVEMBER
          CHARLY VICTOR YANKEE DELTA ALFA PAPA OSCAR SIERRA FOXTROT
          TANGO KILO WHISKY ROMEO GOLF)
```

Fig. 22. Extended example scenario: CBH and Prior Hypothesis of contact 7.

```
KEE Typescript Window
CANDITATE? contact.7

HYPOTHESIS? juliett

(5 5)
MORE? (Yes or No) yes

HYPOTHESIS? echo

(0 95)
MORE? (Yes or No) yes

HYPOTHESIS? (complement echo soviet-sub)
(5 100)
MORE? (Yes or No) yes

HYPOTHESIS? diesel

(5 100)
MORE? (Yes or No) no

SendMessage value: NIL
CANDITATE? c1.10

HYPOTHESIS? juliett

(30 100)
MORE? (Yes or No) yes

HYPOTHESIS? false-contact

(0 20)
MORE? (Yes or No) no

SendMessage value: NIL
```

Fig. 23. Extended example scenario: Dempster/Shafer inquiry.

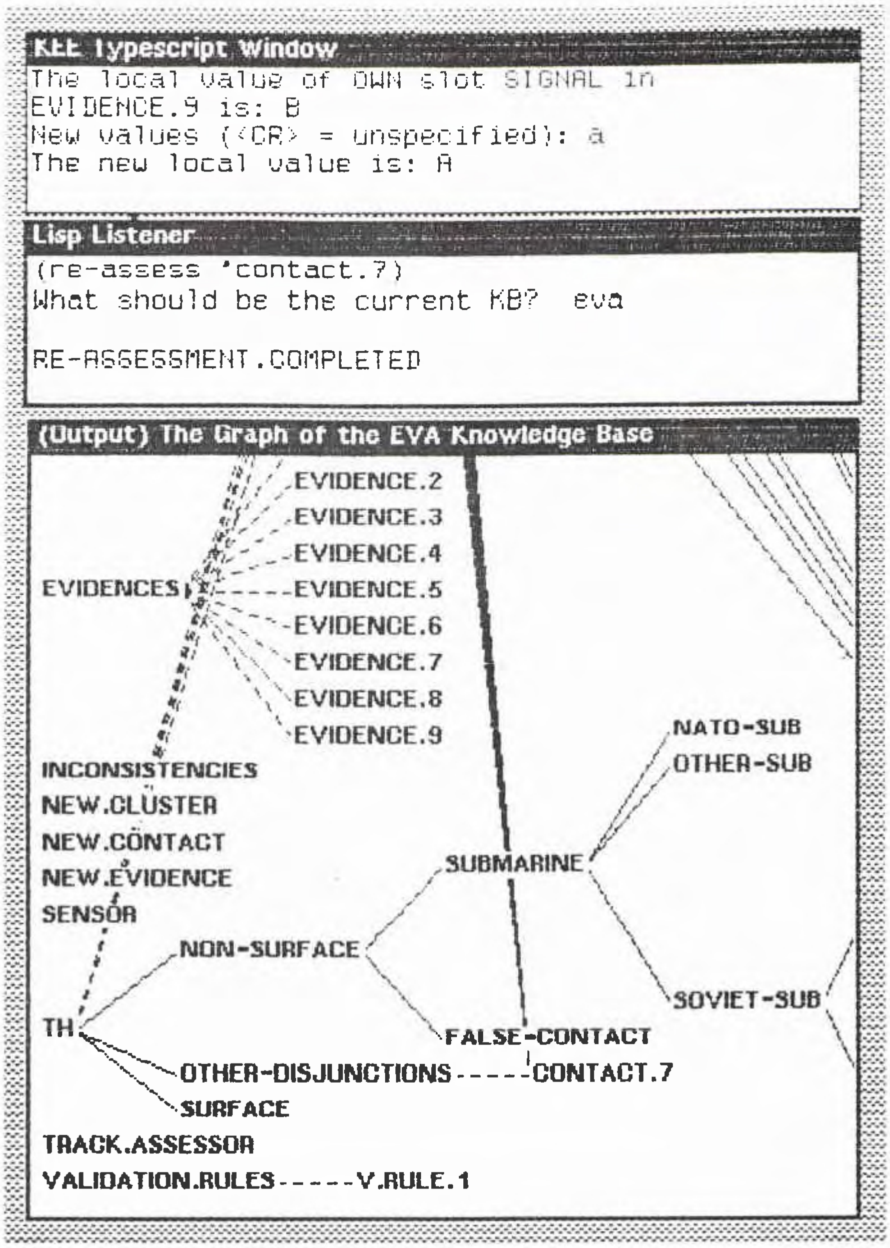

Fig. 24. Extended example scenario: reassessment of a contact.

```
(Output) The P.RULE.C Unit in EVA Knowledge Base
 Values: (IF
           (AND (GREATERP (THE SPEED OF NEW.FACT) 20)
            (LESSP (THE SPEED OF NEW.FACT) 36)) THEN
           (THE EVALUATION OF NEW.FACT IS
            (QUOTE ((NUCLEAR 60)))))

The P.RULE.D Unit in EVA Knowledge Base
 Values: (IF (GREATERP (THE SPEED OF NEW.FACT) 35) THEN
           (THE EVALUATION OF NEW.FACT IS
            (QUOTE (((COMPLEMENT SUBMARINE) 90)))))

The P.RULE.A Unit in EVA Knowledge Base
 Values: (IF (THE EXPOSURE OF NEW.FACT IS SNORKEL) THEN
           (THE EVALUATION OF NEW.FACT IS
            (QUOTE ((DIESEL 80) ((COMPLEMENT DIESEL TH) 20)))))

The P.RULE.B Unit in EVA Knowledge Base
 Values: (IF (THE EXPOSURE OF NEW.FACT IS SNOOP-SLAB) THEN
           (THE EVALUATION OF NEW.FACT IS
            (QUOTE ((JULIETT 80)))))

The P.RULE.E Unit in EVA Knowledge Base
 Values: (IF (THE EXPOSURE OF NEW.FACT IS SNOOP-TRAY) THEN
           (THE EVALUATION OF NEW.FACT IS
            (QUOTE (((COMPLEMENT JULIETT SOVIET-SUB) 95)))))

The SIGNAL.RULE Unit in EVA Knowledge Base
 Values: (IF (THE SIGNAL OF NEW.FACT IS B) THEN
           (THE EXPOSURE OF NEW.FACT IS SNOOP-SLAB))

The SIGNAL.RULE.E Unit in EVA Knowledge Base
 Values: (IF
           (OR (THE SIGNAL OF NEW.FACT IS A)
            (THE SIGNAL OF NEW.FACT IS C)) THEN
           (THE EXPOSURE OF NEW.FACT IS SNOOP-TRAY))
```

```
(Output) The EXTERNAL.FORM Slot of the V.RULE.1 Unit
 Values: (IF
           (GREATERP
            (ABS (THE COURSE.CHANGE OF NEW_CLUSTER))
            1.57) THEN
           (THE VALIDITY OF NEW_CLUSTER IS
            (* 0.1 (THE VALIDITY OF NEW_CLUSTER))))
```

Fig. 25. Rules.

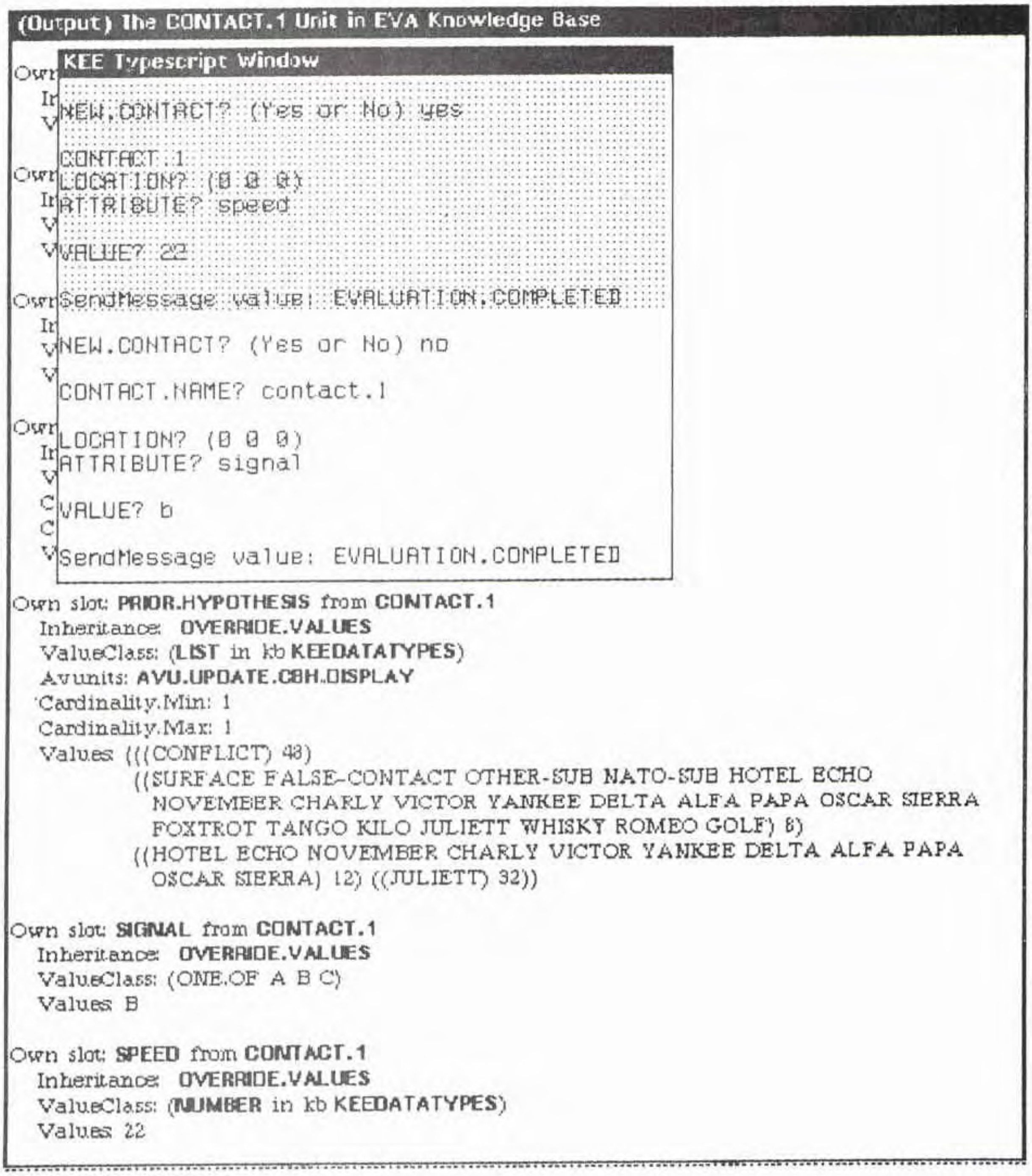

Fig. 26. Case of conflicting evidences.

Report no. changed (Mar 2006): SR-123-UU

Initial Distribution for SR-123

Ministries of Defence	
JSPHQ Belgium	2
DND Canada	10
CHOD Denmark	8
MOD France	8
MOD Germany	15
MOD Greece	11
MOD Italy	10
MOD Netherlands	12
CHOD Norway	10
MOD Portugal	2
MOD Spain	2
MOD Turkey	5
MOD UK	20
SECDEF US	68

NATO Authorities	
Defence Planning Committee	3
NAMILCOM	2
SACLANT	10
SACLANTREPEUR	1
CINCWESTLANT/ COMOCEANLANT	1
COMSTRIKFLTANT	1
CINCIBERLANT	1
CINCEASTLANT	1
COMSUBACLANT	1
COMMAIREASTLANT	1
SACEUR	2
CINCNORTH	1
CINCSOUTH	1
COMNAVSOUTH	1
COMSTRIKFORSOUTH	1
COMEDCENT	1
COMSUBMED	1
COMMARAIRMED	1
SHAPE Technical Centre	1
CINCHAN	3
PAT	1

SCNR for SACLANTCEN	
SCNR Belgium	1
SCNR Canada	1
SCNR Denmark	1
SCNR Germany	1
SCNR Greece	1
SCNR Italy	1
SCNR Netherlands	1
SCNR Norway	1
SCNR Portugal	1
SCNR Turkey	1
SCNR UK	1
SCNR US	2
SECGEN Rep. SCNR	1
NAMILCOM Rep. SCNR	1

National Liaison Officers	
NLO Canada	1
NLO Denmark	1
NLO Germany	1
NLO Italy	1
NLO UK	1
NLO US	1

NLR to SACLANT	
NLR Belgium	1
NLR Canada	1
NLR Denmark	1
NLR Germany	1
NLR Greece	1
NLR Italy	1
NLR Netherlands	1
NLR Norway	1
NLR Portugal	1
NLR Turkey	1
NLR UK	1

Total external distribution	251
SACLANTCEN Library	10
Stock	19
Total number of copies	280