IN DEPTH EVALUATION OF ERATO TOOLS : TOWARDS AN OPERATIONAL PROTOTYPE

Marcel Leroux
CENA,
BP 4005
31055 TOULOUSE CEDEX
France
leroux@cena.dgac.fr

Keywords : cognitive engineering, air traffic control, decision making, human-centred design.

1. INTRODUCTION

The cognitive engineering project, ERATO (En-Route Air Traffic Organiser), has been presented during the first FAA Europe ATM R&D Seminar in June 97. In this paper we present the results of a six month in depth evaluation campaign.

Central to ERATO is a cognitive model of air traffic controllers which elicits the mental mechanisms and cognitive resources that they use to face the demand, and how these resources decay under time pressure, stress and fatigue, so as to provoke error. Thus it enables us to point out bottlenecks in data-processing and in decision-making processes. The functional specification as well as the interface specification of the decision aids are made in reference to this cognitive model.

2. THE COGNITIVE MODEL

2.1 Fundamental mental mechanisms

Four kinds of mental mechanisms are identified, whether they are involved in :

- the management of the physical process (i.e. controlling aircraft),
- co-operation between controllers working at the same control position,
- interface management,
- the management by the controller of his own cognitive resources.

Controllers have to make decision in a dynamic and risky environment. Tasks and objectives are not well defined and evolve very fast. Their definition is a very important part of a controller's activity. Risk is so important that the controller has to guard against errors of all the actors (including himself, the other controllers, the pilots and all the machines) in the system.

They have to process data that depend on the time factor for their value, but also for :

- Their availability. All the data necessary to make a clear assessment of the situation are not available at a given time; some of them may be completely unobtainable. Very often the controller must take decisions in a state of partial ignorance. this is done using default reasoning techniques associated with the monitoring of "sentry parameters" that guarantee that the effective behaviour of any aircraft remains consonant with the default one.
- Their accuracy. When observing air traffic controllers, we found that they spend a lot of time and cognitive resources in eliminating ambiguity. Fuzziness and uncertainty makes

necessary the use of fuzzy reasoning associated with relevant parameters monitoring processes.

Default reasoning and fuzzy reasoning have in common to produce defeasible conclusions. Conclusions are revised through out the acquisition of data in the environment. We can describe the very tight relations between cognitive activity and data acquisition from the interface. For the time being, data presentation is technology driven and needlessly bulky.

- Their flow. Controllers must adapt to sudden transitions between situations characterised by lack of data to those involving data overflow. Their mental and perceptive activities may be interrupted at any time by pilots' calls, telephone calls etc. This creates a very dense memorisation activity. The danger associated with forgetting a conflict situation, as well as the great number of data-acquisition processes and the rapid changes in the status of these processes as time goes by, make memory management a very demanding task for the controller. We observed three major problems regarding memory :
- temporal deadline memorisation,
- frequent goal-shifting and context-restoration,
- memory fading with time.

2.2 Co-operation

Most of the previous tasks can be performed by the two controllers, successively or in parallel. Mental mechanisms involved in co-operation are an essential part of the model. Efficient co-operation between the two controllers relies on three factors:

- they must share the same skills, knowledge and training,
- they need to have consonant representations of effective traffic requirements,
- they must have simultaneously available cognitive resources to exchange information.

2.3 Cognitive resource management

All these processes are highly interactive. In demanding situations they may severely compete with and constrain each other. The controller has to attribute limited cognitive resources to these multiple relevant goals. During previous experiments we verified that the difficulty of a problem is not intrinsic to the problem but mainly depends on the context in which this problem has to be solved.

The operator has to attribute limited cognitive resources to these multiple relevant goals. Central for the operator is the need to select the right task among several pending ones and to perform it in the right time span. Resource management depends, among several factors, on the short-term-taskload assessment : the number of tasks to be performed with their time span, their status, their technical difficulty, their critical dimension, and their coupling. The goal-switching difficulty depends on the number of shifts and on the number and complexity of context restorations.

3. THE FUNCTIONAL SPECIFICATION OF DECISION AIDS

3.1 Guidelines

The problem that we have to consider is how to assist the controller in processing information. The functions and their HMI must support cognitive activity under risk and time pressures. They must also support efficient co-operation between controllers working on the same sector suite.

The assessment of bottlenecks in data processing and in real time decision making processes shows that, given a working context, human performance has a validity domain. The aim of Erato is to expand the limits of the validity domain of human resources. Thus, automation is no longer thought of as a means of progressively expelling human operators from the decision-making process. It becomes instead a means of improving human performance, either by magnifying the efficiency of cognitive resources or by improving cognitive resource management.

For the next decade, the nature of the data to be processed by the controllers will not change significantly. So, all the mechanisms that are inherent in the nature of data to be processed must be preserved in the new environment. Prior to this specification is the definition of the co-operative working method in an electronic environment :

- What do controllers actually need to build an effective mental representation of traffic in a co-operative way and under time pressure ?
- How should the system support decision making processes and co-operation between controllers.

This strong interaction between the specification of the tools and the definition of the working method is critical throughout the iterative process of defining the Joint Men-Machine System (JMMS).

However, the cues that trigger mental processing are associated with the physical environment, so they will disappear. It is necessary to make sure that the new environment will enable the operator to obtain a relevant set of efficient cues.

Cognitive tools can be specified :

- either to improve the efficiency of cognitive resources, (or to save these resources),
- or to manage them in a more efficient way.

3.2 Improving the efficiency of cognitive resources

This can be achieved using information filtering techniques or by providing an interface which comes as close as possible to satisfying cognitive needs. Erato provides two kinds of filtering functions.

- Task-driven data presentation

The electronic interface makes it possible to present only the relevant parts of the raw data set so as to show relevant data in a way that enable the controller to perform the task more efficiently. Task driven filtering functions include a list of flights organised according to their cleared flight level, a list of flights organised according to their exit flows.

- Problem-driven data presentation

The aim of problem-driven information filtering is to reduce the number of aircraft to be considered at one time. By splitting a very demanding situation into several subsets of aircraft, we can expect that the controllers will be able to process these subsets of aircraft very efficiently. The problem-driven filtering function can be triggered from the radar display, the flight integration window (FIW) or any of the lists of flights.

This function can be applied to any aircraft known by the system. It dims all the aircraft that may in no way interfere with the reference aircraft selected by the controller. The set of aircraft that remain highlighted includes all the aircraft that may be conflicting with the reference aircraft, and all the aircraft that may interfere with the resolution of potential conflicts detected by the controller. This function includes two what-if functions (to simulate the effect of level changes or of direct tracks).

- Interface specification

Using the cognitive model, the interface must be specified as an information display, a support to mental processing, a source of cues that may trigger mental processing, a support to inputs. It is also central as a support to co-operation between controllers.

3.3 Improving the management of cognitive resources

The reminder consists of a specific window where each problem will be tagged. A problem is defined as a conflict situation involving two or more aircraft. The labels are positioned according to the urgency of resolution. The problems that are proposed by the machine must be considered as a draft by the controller. He can modify the labels so as to adapt them to the effective needs of the executive controller, and particularly, he can adjust the target resolution time. At resolution time, the relevant aircraft are highlighted on the radar display.

The display of the relative urgency of problems should enable the controller to avoid wasting cognitive resources on non-urgent and unimportant tasks while the short term situation decays. In normal operations, this should allow the controller to objectively manage all cognitive resources and avoid tunnel vision errors.

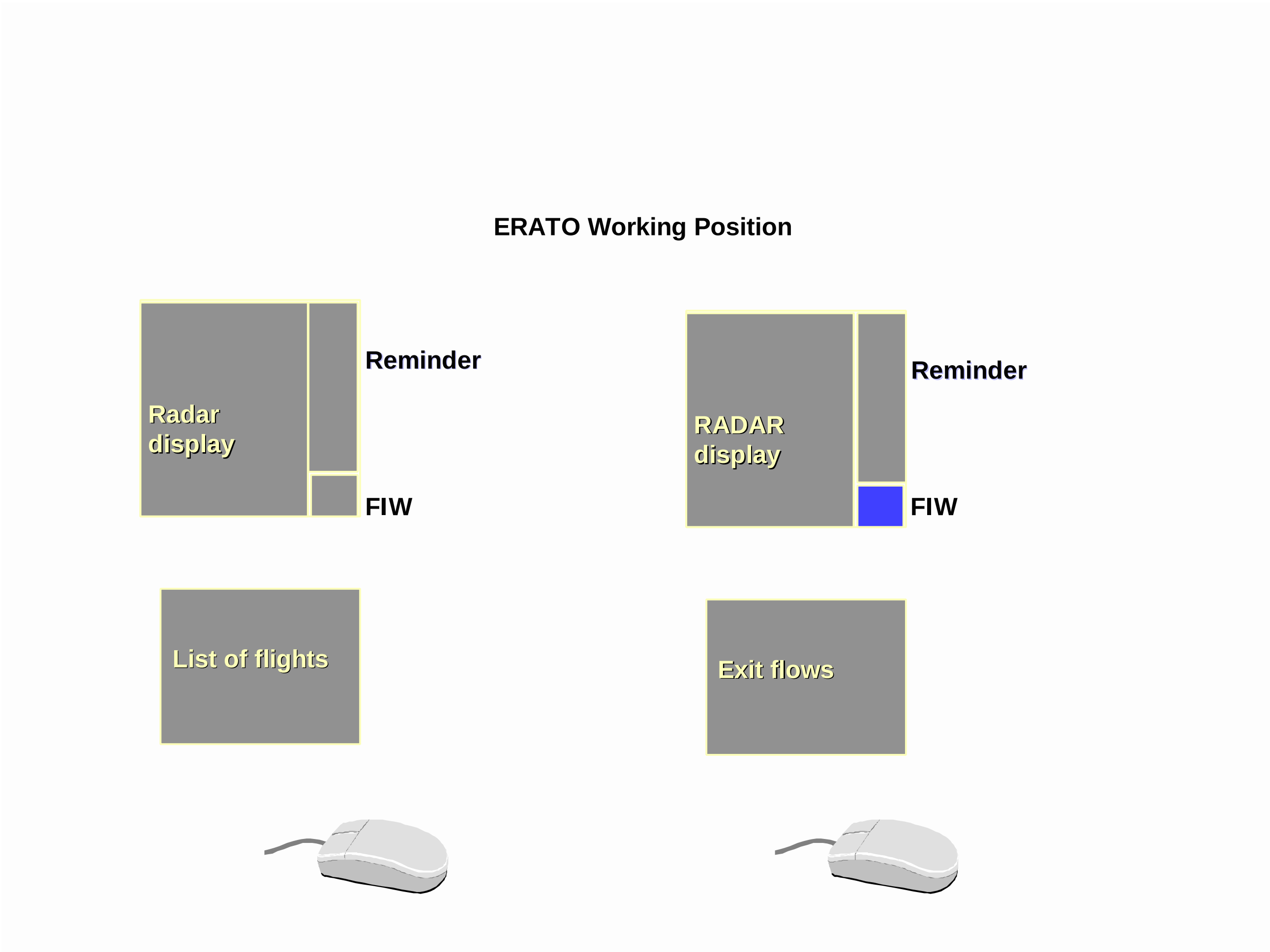

Figure 1

The aim of the reminder is to show the two controllers what the traffic requirements are and their urgency. Thus, it should enhance co-operation between them. It should be used by both controllers as a safety net based on intentions of action.

Erato uses the ODS-France sector suite. The EC position (left) and the PC position (right) are quite similar. For both controllers, the Sony screen includes three windows : the radar display, the reminder and the Flight Integration Window (FIW). The task-driven lists of flights may be displayed on each 19" screen.

4. EVALUATION OF THE JOINT MAN-MACHINE SYSTEM

4.1 The role of the cognitive model in defining the experimental protocols

The cognitive model provides a guideline for evaluating the Joint Men-Machine System (JMMS). It enables the transformation of high level validation requirements into relevant criteria to test the JMMS. It determines which aspects of the machine or of the man-machine interaction must be verified closely so as to guarantee an effective performance of the whole system or to prevent error. One can then determine or assess the gains in these aspects.

We must verify that the new JMMS preserves the sources of good performance and really improves the weak points from both a safety and a capacity point of view. The man-machine interaction must enable operators to exercise all the mental mechanisms necessary to build the relevant mental representation of the traffic to be controlled. It must also enable them to cooperate in an efficient way. And at last, the HMI must provide the relevant cues, the relevant data and enable efficient inputs into the system.

The experiments should also enable one to describe how the JMMS evolves, how the bottlenecks in the operator's activity evolve, disappear, decay or are created; what kind of problems are solved and created by the new system.

Finally we must assess the performance of the joint man-machine system, and answer questions such as:

- Does it improve the global performance, from the capacity and safety points of views?
- Does it enable them to work in a more efficient and creative way?
- How will it modify their activity?
- Does it produce a loss of vigilance or of skill?
- What are the consequences for training?

The data will enable us to evaluate the JMMS along seven dimensions :

1. Training,
2. Validation of the algorithms,
3. Evaluation of the JMMS from the cognitive activity point of view : how do the functions and the displays support the cognitive activity of each controllers? Do they remain "ahead of traffic" in very demanding situations ? Are they able to compensate for any unexpected situation? Which kind of error appear? Can they detect these errors and deal with them?
4. Evaluation of the JMMS from the co-operation point of view : is co-operation between controllers more efficient ? How is it supported by the system?
5. Usability of the HMI from both the cognitive and ergonomic points of views : does it show the right data and the relevant cues ? Are these data easy to interpret ? etc.
6. Evaluation of the JMMS from the fatigue point of view. The aim of these tests was to verify that the controllers were able to control a traffic corresponding to the present peak of traffic, but during 8 hours.
7. Acceptability by the controllers of the automation philosophy, the HMI, the working method.

This paper describes only points 3, 4 and 5.

4.2 The role of the cognitive model in training

The evaluation of training is a very important part of the trials, as it is essential to make sure that people can get acquainted with the new system in a short period.

To optimise training, we had to anticipate the problems that would be encountered by controllers, so as to focus on these points. To do so we used the cognitive model in order to determine plesiomorphic features and apomorphic features in controller activity. Plesiomorphism mainly concerns the physical process management (skills to detect and resolve conflicts), while apomorphism concerns all the physical and cognitive activity linked with the interface and co-operation.

In the same way, present co-operative activity is supported by the interface. A pure electronic data transfer proves to be inefficient in demanding situations: controllers have to learn how to preserve the rich multimodal co-operation in an electronic environment, how to make the tools improve this multimodal activity rather than sterilise it.

Training includes a static phase (3 days) followed by a dynamic one (6 days).

During the static phase, the behaviour of the machine is explained :

- what are the functions that are available on the interface,
- what are the data that are displayed,
- the rationale behind it,
- how to use these data and these functions to build a relevant mental picture of the traffic,
- what are the relevant cues on the interface? Where to pick them ? How to interpret them ?

Then controllers have to practise on increasingly-demanding traffic samples, so as to restore an efficient visual scanning on the interface. All the cues they are used to, and all the associated reflexes, depend on the present interface : they are no longer available. So controllers must learn how to make efficient use of all the visual cues provided by the new system.

The initial results show that all the controllers succeed in being really efficient before the end of the two-weeks training-period, whatever their age and length of service as Full Performance Level controllers. Table 1 summarises the planning for each team.

Week	Days	Activity
1	1 to 3	Static training
	4-5	Training on traffic simulations
2	1 AM	Test on static training
	1-5	Training on traffic simulations
	5 PM	Debriefing on training
3	1-2	Activity and co-operation trials
	3	Usability trials
	4	Fatigue trials
	5	Debriefing

Table 1

4.3 Experimental protocol

4.3.1 The activity and co-operation trials

For these two days trials, the experimental protocol was based on the evaluation of the performance of controllers in demanding or very demanding situations.

The experiments involved eight teams of two controllers. Each team was confronted with two traffic simulations, the average duration of a simulation was 90 minutes. A video record of the controllers' activity and a computer record of all the actions on the system was made.

Figure 2 shows the main streams of data and information. Some pieces of information (arrows 1 and 2) are displayed automatically by the system, without any action of the controller (warnings, new flights, new problems, etc.).

Arrows 3 and 4 represent the actions of the controllers, either the input or the consultations on the HMI.

Arrows 5 and 6 show the communications between the pilots and the executive controller. In fact these VHF communications are also heard by the planning controller.

Arrows 7 and 8 are for the co-operation between both controllers, which include oral communication and gestures.

Arrows 9 and 10 represent the telephone communications between the planning controller and the neighbouring sectors. The executive controller may hear what is said by the planning controller.

All these data are recorded. The experimental environment enable to synchronise these different sources of data so as to review a simulation in a very realistic way. Immediately after each evaluation simulation, this provide a support for interviewing the controllers in a semi directed way on their own activity, the way they have controlled the traffic, how they have used the tools, etc. We have got 16 deferred interviews, named "activity interviews" in this paper ; the average duration of them is about 4 or 5 hours. The third day was dedicated to directed interviews and trials on the usability of the interface (the "usability interviews"). The duration of the usability interview was about 5 hours with each controller. So we have got about 150 hours of interviews, which represents a very important source of data.

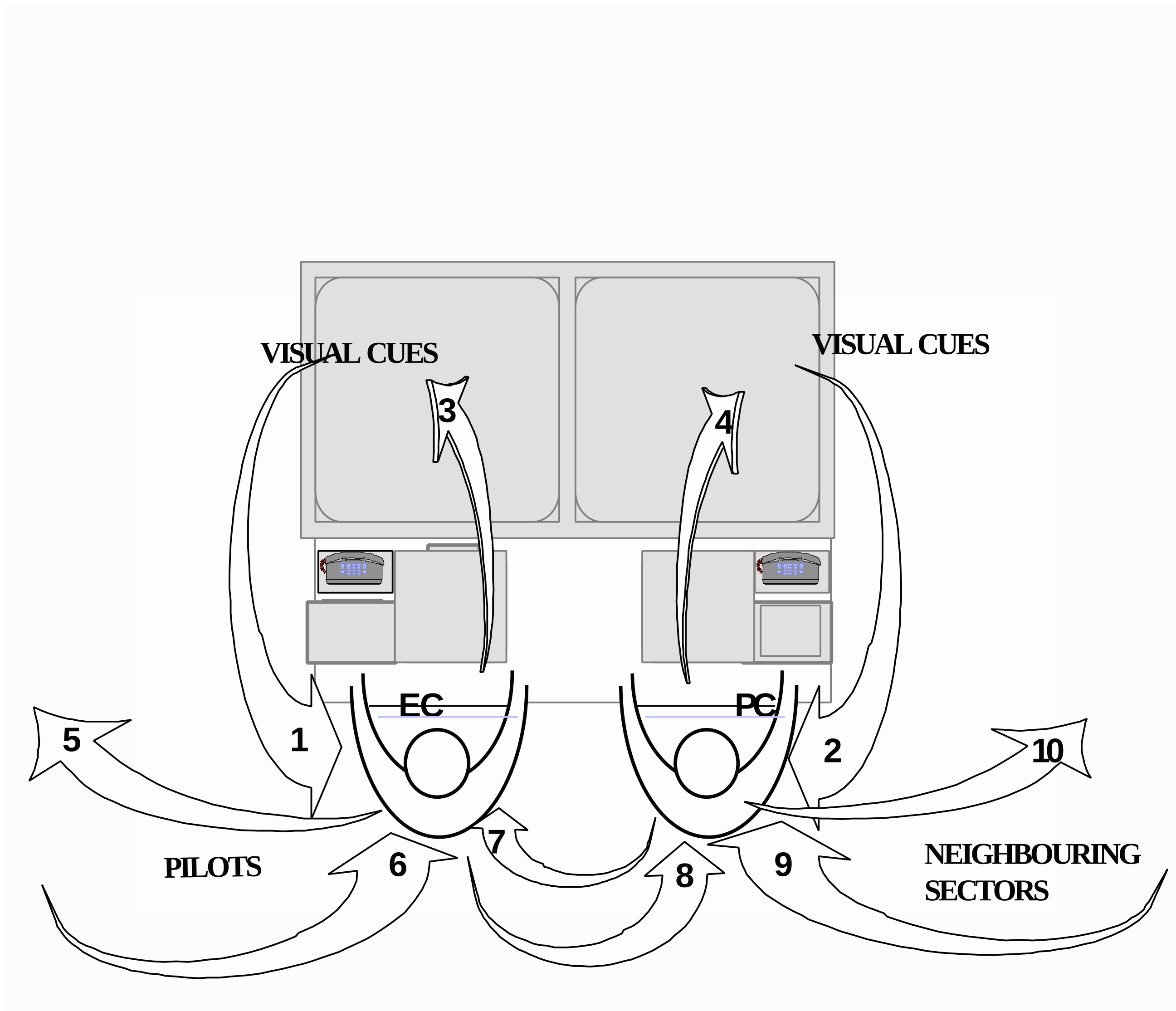

Figure 2

4.3.2 Description of data

Table 2 shows the number of events that occurred during 4 traffic samples of 90 minutes on 4 different control sectors. They give an idea of the density of information that the controllers have to process.

The line "system" is the number of events corresponding to the automatic display of information by the system. This does not include the radar display updating.

The "EC actions" and "PC actions" lines are the number of actions on the system made by the controllers.

The unit of speech is any non interrupted oral act : it may be a full sentence or a single word.

The planning controller is explicitly or implicitly concerned by all these data except most of the EC actions on the system (he is obviously concerned by the inputs). The executive controller is explicitly or implicitly concerned by all these data except most of the PC actions on the system.

The result is that a new more-or-less sophisticated pieces of information is available every one second and a half.

	traffic sample 1 (90 minutes)	traffic sample 2 (90 minutes)	traffic sample 3 (90 minutes)	traffic sample 4 (90 minutes)
System	565	673	632	745
PC actions	2168	2403	1692	1315
EC actions	1255	1837	1792	1001
PC speech	268	237	363	369
EC speech	627	596	596	596
Pilots speech	250	299	292	424
Neighbours	11	33	63	77
total for PC	3889	4241	3638	3526
total for EC	2965	3642	3675	3135

Table 2

4.4 Data encoding

All these data are encoded using Mac Shapa software predicates :

- two classes of predicates enable to encode the data that are automatically displayed by the system (one for the executive controller's HMI and another one for the planning),
- two classes of predicates enable to encode the actions of each controller on the interface (input, queries)
- a class of predicates enable to encode the activity of controllers (speech, gestures).

Adapted queries enable to restore both the activity and the imbrication between this activity and the use of the interface. It enables to understand what are the decision-making processes and how the interface support these processes.

4.4.1 Data that are displayed automatically on PC and EC's HMI

These data are coded by means of 6 predicates. For example, the predicate S_PBA indicates the display of a new problem in the reminder. This predicate is coded under the following form :

S_PBA($dest, $pb, $pb_id, $hsr, $uts),
where $dest is the target controller's HMI,
$pb is the name of the problem,
$pb_id is the problem system identification,
$hsr is the associated resolution time,
$uts is the list of relevant aircraft

For example the following predicate "08:23:11:00 S_PBA(PC, 2 CROSSING CTL 330, 5, 08:35:57:00, SWR801R SAB965)" means that at time 08:23:11:00 the system has displayed a label in the reminder of the PC, concerning two aircraft crossing over the fix CTL at level 330, the system identification of this problem is 5, the resolution time is

08:35:57:00. The aircraft callsigns are SWR801R and SAB965.

These predicates are automatically generated by the simulation system.

4.4.2 EC and PC's input and data consultation

These data are coded by means of about 40 predicates. For example, the predicate FILT indicates that a controller has asked for a filtering. This predicate is coded under the following form :

FILT($orig, $cs, $afl, $normal, $eai, $object, $fiw) where $orig is the controller who has asked for the filtering,

$cs is the call sign of the reference aircraft,

$afl is the flight level,

$normal indicates if it is a "normal" or simulated filtering,

$eai shows the list of the aircraft belonging to the filtering,

$object indicates which representation of the flight has been used to ask for the filtering,

$fiw indicates whether it is the first filtering on this aircraft or not.

For example the following predicate "11:53:27:00 FILT(EC, TA591VR, 330, normal, CRX821 AFR473, FIW, first)" means that at time 11:53:27:00, the executive controller has made a filtering on TA591VR, at level 330. It was a normal filtering (ie not a simulated one). The highlighted aircraft were CRX821 and AFR473. The filtering has been made from the Flight Integration Window (FIW). It was the first filtering on this aircraft.

These predicates also are automatically generated by the simulation system.

4.4.3 Speech encoding

During these trials we have recorded 24 hours of traffic simulations. The transcription of real time conversations between both controllers, of pilot-controllers VHF messages, of telephone messages with neighbouring sectors, need to be encoded under the same formalism of predicates, so as to be easily combined with the HMI predicates.

The complexity of this encoding comes from the fact that the objects that are referred to in oral communication are often complex : they may refer to elementary data, "what will be the exit level for this aircraft ?", or to cognitive activity. For example a sentence like : "I'm not sure that I'll have to vector..." may refer to an ambiguity elimination process on the diagnosis (conflict or not) or on the resolution frame (radar vectoring vs speed reduction or level change).

The name of the predicate is the speaker, (EC, PC, pilot, neighbour sector).

The first argument refers to the media (speech, speech addressed to the colleague, VHF, telephone)

The second argument refers to different contexts (workload, interruption management, traffic management, HMI management, micro incidents on co-operation, etc.).

The 3^{rd} and 4^{th} argument refer to the aircraft mentioned in the intervention, and when it is clearly mentioned, to the cluster of aircraft to which this one belongs

The content of the intervention may be explicit or implicit (5^{th} argument).

The goal of the intervention or of the communicative act may refer to an action or may be declarative, including a request of information or providing additional pieces of information (6^{th} argument).

The intervention may initialise an exchange or may be reactive (7^{th} argument).

The speech may explicitly refer to a cognitive activity. This is encoded in the five following arguments. The 8^{th} argument encodes the kind of cognitive activity : default reasoning, ambiguity elimination process, memory refreshment, etc. The intervention may refer to an "abstract object", which may be a diagnosis, a resolution frame, etc. (9^{th} argument). The tenth parameter enables to instanciate this abstract object. The eleventh one codes the author of the "abstract object". The twelfth argument mentions the appreciation of the speaker on this abstract object : agree, disagree, may be, perhaps : it is the way of coding how people exchange "abstract object" and progressively assess its value. When the controller explicits which tool/function has supported this cognitive activity, this function is coded in the 13th argument.

The 14th argument encodes the tasks that result from the exchange, when these tasks are explicitly mentioned by the speaker.

Encoding one hour of speech needs about 8 man*days.

This encoding enables to formalise what is said by the controllers. It is a way of accessing a little part of their real time cognitive activity. But unfortunately, cognitive activity is not necessarily verbalised. For example one of the consequences of plausible inference is defeasible conclusions. At a given time the controller may evoke a resolution frame and a few minutes later deliver a clearance which is different. This proves that he has changed his mind but he may not necessarily have verbalised this change. This in-depth analysis cannot be made by hand, but must be made automatically by using "speech predicates". So encoding the speech is a necessarily step in the processing of data.

A larger part of the cognitive activity can be inferred by means of "activity queries". These queries enable to combine speech predicates with the predicates describing the actions on the HMI and the predicates associated to the data that are automatically displayed on the HMI. The activity queries enable to detect instances of the cognitive mechanisms, such as default reasoning, ambiguity elimination, interruption management, focus of attention, memorisation, etc. It also enables to describe very precisely these mechanisms, to describe how the functions support this activity, to analyse it and finally to assess the usability of the interface in a very precise way.

5. RESULTS

5.1 Statistics

For the time being, the statistical analysis is decomposed into 254 questions, about :

- numbers of events, either elementary events (for example, the number of filtering by the EC) or complex events (for example, the number of aircraft for which the first action on the HMI has been made by the executive controller before the planning controller and via the reminder).
- mean, maximum, minimum durations of use of any functions. These durations can correspond to elementary or complex events.
- cycles and sequences of use of functions.

The following table shows a little subset of these statistical data, representing four questions amongst 254, and for only 4 teams of two controllers. It corresponds to eight different demanding traffic samples of 90 minutes each. Each team has controlled two different traffic samples. The controllers have switched, so as to play the two roles of executive and planning controllers. For example T3-1 represents the first controller of the third team. "E" and "P" represent the data corresponding to the traffic simulation where he has been respectively executive and planning controller. The durations are expressed in seconds.

5.2 Interpretation

5.2.1 Number of filtering on problems in the reminder

The first line of table 3 shows that all the controllers, but controller 2 of team 2, use much more the filtering from the reminder when they act as a planning controller than when they are executive controller. This must be confirmed by the "usability interviews" so as to explicit the rationale behind that.

Two exceptions should be detailed : the second controller of team 2 has filtered the same number of times when executive or planning (77), the second controller of team 4 has filtered only twice when he acted as an executive controller.

	T1-1	T1-2	T2-1	T2-2	T3-1	T3-2	T4-1	T4-2
Number of filtering on problems	P : 89 E : 14	P : 57 E : 22	P : 107 E : 38	P : 77 E : 77	P : 96 E : 37	P : 108 E : 30	P : 51 E : 21	P : 55 E : 2
Number of filtering on aircraft	P : 90 E : 129	P :125 E : 118	P : 105 E : 85	P : 69 E : 103	P : 111 E : 95	P : 152 E : 193	P : 86 E : 108	P : 111 E : 119
Average duration of filtering on problems	P : 23.55 E : 15.28	P : 18.68 E : 10.09	P : 15.60 E : 13.34	P : 9.66 E : 4.58	P : 8.33 E : 5.81	P : 12.09 E : 9.46	P : 25.81 E : 21.76	P : 20.9 E : 7
Average duration of filtering on aircraft	P : 13.47 E : 8.87	P : 9.27 E : 9.77	P : 9.85 E : 9.69	P : 7.23 E : 4.99	P : 8.35 E : 6.36	P : 7.93 E : 7.02	P : 6.09 E : 9.55	P : 22.82 E : 16.82
Maximum duration of filtering on problems	P : 156 E : 46	P : 83 E : 44	P : 101 E : 81	P : 82 E : 68	P : 45 E : 29	P : 145 E : 40	P : 135 E : 65	P : 134 E : 11
Maximum duration of filtering on aircraft	P : 139 E : 62	P : 48 E : 84	P : 39 E : 63	P : 155 E : 51	P : 49 E : 23	P : 60 E : 44	P : 55 E : 90	P : 172 E : 140

Table 3

5.2.2 Number of filtering on aircraft

For each controller, the comparison between lines 1 and 2 of table 3 shows that when the controller acts as an executive controller, he uses much more the filtering on aircraft than the filtering on the reminder.

5.2.3 Average duration of filtering on problems in the reminder

For each controller, the average duration of filtering on the reminder is longer when he is the planning controller than when he acts as an executive controller.

To summarise, when a controller acts as a planning controller, he uses the reminder filtering much more than when he is an executive controller, and the average duration of each use is significantly longer.

This is relevant with the goal assigned to each of these tools : filtering on the aircraft is a good support to traffic analysis, the main role of the reminder is to provide a global view of the situation. Reminding there is a pending problem is easily done at a glance on the reminder. Analysing how to resolve it can be done either by filtering on the reminder or on a relevant aircraft, but the best way to memorise an aircraft remains the filtering of this aircraft.

From these data we can suspect that the reminder is not used in the same way by the planning and the executive controllers. The first one must work out the reminder, so as to meet the effective needs of the executive controller, while the second is more a "consumer" of the reminder. These assumptions are relevant with the working method that have been proposed. They must be verified through an in-depth analysis of co-operation between controllers and of their cognitive activity.

5.2.4 Minimum durations

These minimum filtering durations are not mentioned in table 3 because all of them are almost equal, and lower than 2 seconds.

Several hypothesis may be evoked to explain these very short durations and must be checked. It may be due to :

- input error : the controller may click on the wrong aircraft and cancel immediately, or
- refreshing memory activity : this must be cross-checked with concomitant speech, and with some of the results of usability trials. During these trials, one of the experiments consisted in showing the controllers, for one second, a complex Erato display (including a filtering and various warnings) so as to evaluate how many items a controller was able to detect in one second.
- interruption management : the controller wants to analyse an aircraft and is interrupted by another pilot call or phone call. In this case, it is very important to analyse how the controller recovers from this interruption, what are the functions that are used to memorise this interruption, etc.,
- an instance of the monitoring process to eliminate ambiguity on the diagnosis or on the choice of the resolution frame. We need to cross-check with real time activity.

5.2.5 Maximum durations

The figures that appear in lines 5 and 6 must be related to the density of information flows as mentioned in table 2. They are enormous. Several hypothesis can be considered and must be closely verified.

1- It may be a support to "normal" activity :

- assessing the situation in complex situations,
- co-operation,
- micro incidents management.

2- Or it may be due to incidents or errors :

- does it correspond to high workload or low workload periods ?
- is it due to tunnel vision problems, indicating a failure of the mechanisms of control of the focus of attention ?
- is it due to an error in interruption management (in this case, when interrupted the controller should have dissociated the management of the interrupting task and the use of the interface) ?
- is it a simple forgetting to reset the interface ?

A deeper analysis must be carried out on the causes of all the occurrences of long durations of use, the context in which they happened, the consequences that can be observed either on the activity, or on the co-operation.

5.3 Cross-checking with controllers' opinions

The usability interviews can provide a rationale behind some specific use of the interface and explain some differences in the use of the functions.

For example, we have to understand why controller T4-2 has used the filtering on problems only twice. Does it correspond to a deliberate way of using the tool, corresponding to a rationale ? or is it pure coincidence ?

During the usability interview, this controller explained that reading the label of a problem in the reminder (that means without any click on it to get the associated filtering) is often sufficient to assess the nature of the problem and even to trigger the relevant resolution frame. In this case we must distinguish the use of a tool from its manipulation. This assertion must be carefully verified by analysing the real time activity of this controller.

From the usability point of view, the consequence is that the pieces of information available in the reminder label are essential for that controller : the name associated to the problem, the location and level of the conflict, the list of interfering aircraft.

Though interesting, the cross-checking between statistics and usability interviews is not sufficient, for two reasons. On this point the controller T4-2 clearly explains the rationale behind his behaviour, but in some cases there may be a gap between the way people say how they use a function and the raw results of statistics.

And people are unable to explain all about the context in which they use the interface, specially on the cognitive activity. In this specific example of controller T4-2, we must also verify the nature of co-operation between controllers, by using the activity queries : this way of using the reminder may implicitly be supported by a specific way of co-operating. These queries will be described during the oral presentation.

5.4 Describing and analysing the activity

Analysing the activity is twofold :

At first it enables to interpret features from the statistic analysis, as mentioned above. Statistics enable to raise questions, but cannot provide relevant answers on the effectiveness of the functions and their usability. For example, we cannot interpret the maximum and minimum durations of filtering if they are not replaced inside their cognitive context.

Second, this analysis enables to verify that Erato preserves those mental mechanisms that, being inherent to the nature of data that controllers have to process, are essential in building and maintaining the mental representation of the situation.

In both cases, we cannot be satisfied with a global assessment of activity. The interaction between cognitive activity and the use of the interface is so dense, so symbiotic that we need a very thin granularity if we do not accept to rely on subjective assessment by the controllers to identify the sources of good and poor performance.

This is why the encoding of speech and the use of activity queries are central. This makes possible an automatic research of patterns that are characteristic of cognitive activity. Some of these patterns will be described during the oral presentation.

For each of these patterns, the automatic research enables to answer the following questions :

- What are the cues in the environment (visual cues on the HMI, oral cues, etc.). These cues trigger all the mental processes that enable the controller to build his mental representation of traffic. Building this representation is not a deterministic process, it is guided by the detection and the processing of these cues. So it is critical to verify that the interface provides relevant cues, and that the controller can easily detect and process them.
- How the context (workload, co-operation problems, micro incident management, etc.) modifies this activity.
- How the cognitive activity is carried out, what are the different forms according to co-operation, time pressure, etc.
- Which functions support this cognitive activity ? Which data are used ? On which windows ? Are these data directly available or does the controller need to act on the interface to get them ? When the data are available, are their easy to interpret ?
- Each of these mechanisms may be performed by a controller alone or in a co-operative way. Does Erato preserve the capability to cooperage ? What are the functions that have been used to support co-operation ?
- Can we observe micro incidents, and describe these incidents : how were there detected and fixed up, what are the functions that are involved and have contributed to provoke the incident, which functions have been useful to detect the incident or to recover it.

These cross checking cannot be done manually. This is one of the justifications of speech encoding.

5.5 Consequences

The consequences of these analyses may be to improve the design of the functions, or of their HMI. It will be an irreplaceable source of data to refine the working method, to improve training and to lay the foundations of TRM in an electronic environment.

6. CONCLUSION

Intermediate results will be given during the oral presentation. The informal reactions of all the controllers that attended these trials and the first results show that this model driven way of specifying the future electronic environment is successful. All the controllers that have been involved in these trials declare to be very satisfied with this electronic environment, provided it includes minor modifications on the HMI. All of them succeeded in managing with much more traffic than they could have done using the present tools, and for a much longer time.

Though these cheers are very encouraging, they cannot substitute to a scientific evaluation of the future Joint Men Machine System. This evaluation may seem costly. Its cost is negligible, compared to the global cost of the design of such a new environment. Beyond the appearances, we must make explicit the reasons why the future JMMS works well, the conditions that are necessary for it to be efficient, so as to master the transition between the present system and the future one, and to master the evolutions of the future system.

The filtering algorithms and the reminder algorithms are now encoded by the French technical services (STNA). The evaluation of these algorithms connected to the real system are now on progress. The next major step will be trials in real conditions in very demanding situations by early 2001.

Acknowledgement

This paper could not have been written without the results got by the whole Erato team : Sonia Abdesslem, Nicole Boudes, Marie Christine Bressolle, Corinne Capsié, Anne France Corredor, Lise Mesguen, Robert Parise, Elyeth Tremblay.