



# ERATO: Cooperative Tools Based on Cognitive Engineering

Marcel Leroux

## ► To cite this version:

Marcel Leroux. ERATO: Cooperative Tools Based on Cognitive Engineering. ATM 1997, 1st USA/Europe Air Traffic Management Research and Development Seminar, Jun 1997, Saclay, France. hal-01291546

**HAL Id: hal-01291546**

**<https://enac.hal.science/hal-01291546>**

Submitted on 21 Mar 2016

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

## ***COGNITIVE ASPECTS AND AUTOMATION***

(to appear in "Cognitive Engineering in Aviation" Amalberti & al eds)

***Marcel Leroux,***

CENA, 5-7 Avenue E. Belin,  
BP 4005, 31055 Toulouse CEDEX  
FRANCE

e-mail : leroux@cenatls.cena.dgac.fr

### **1. INTRODUCTION : Automation or cognitive tools ?**

The European Air Traffic Control system (ATC) has entered a deep crisis : this system is unable to meet a tremendous increase in the demand. This is not only the consequence of its inertia ; such inertia is normal for any complex system. Short term measures to optimize the present tools appear to be insufficient : these tools have already reached the limits of their development capability. A wide-ranging discussion, on how to enhance ATC methods and tools, has begun. Very ambitious goals have been set for the future systems : for example the French CAUTRA V project plans to double the capacity of the system by the year 2005 and to significantly increase safety. Numerous ambitious projects exist, but none of them has yet proved its efficiency or even its feasibility. ATC automation is really short of effective solutions.

Obviously, major technology improvements (FMS, Data Link, 4D-Navigation, computational power) must be intensively used. But full automation cannot be a solution, at least for the next two or three decades. Human controllers must remain in the decision-making loop. As automation cannot replace human operators, it must assist them. This is one of the paradoxes of automation, as needed by the ATC system : until full automation has proved to be feasible and efficient will not be proved, i.e. as long as controllers will be needed to make decisions, even occasionally, *it is essential to preserve the controllers' skills*. Whatever the tools that may be designed, human controllers must exercise their skills continuously.

Human operators provide flexibility, the capability to deal with unexpected situations, creativity, and safety, thanks to their ability to compensate for the failures and inadequacies of machines. To preserve these capabilities, we may have to automate "less" than is possible from a purely technological point of view.

But the human operators are also a potential source of error. From this observation and for years, system designers thought that the more human operators could be put on the fringe, the more the risk of error would decrease.

The underlying assumption was twofold :

- the technological advances will be sufficient to significantly improve the performance of the joint man-machine system,
- human operators are flexible enough to adapt to the new working environment, which is in contradiction with the initial goal of expelling humans from the decision-making loop in order to exclude their potential for introducing errors.

In fact technology driven automation adds another kind of difficulty to the supervision of the initial system : the difficulty of understanding the behavior of the automatic systems that partly monitor the system. Thus, it creates additional sources of errors ; the consequences of these errors are much more serious than the previous ones. So it would seem that rather than eliminating human operators with the consequences of depriving the

joint system of major benefits and of increasing the risk of errors, it would be more sensible to design a system which was error-tolerant. Such a system cannot be designed merely by making use of technical advances : we must automate in a different way than suggested by technology alone. Thus the word "automation" seems to be inappropriate, "cognitive tools" or "computer assistance with cognitive tasks" would appear to be more suitable.

The problem that we have to consider is how to assist the controller in processing information : the controller's activity consists in real-time and cooperative data-processing so as to "produce decisions" under risk pressure. Up to now, the objects presented on the interfaces (the paper flight progress strips and the radar display) correspond to the aircraft, while the controller mainly processes the interactions between aircraft. Can computers help the controller to perform this task ? The aim is no longer to show the results of radar tracking, but, to use these results as well as all the technological advances in order to present relations between aircraft in a more manageable way. In this paper the cognitive engineering approach is presented as an alternative to technology-driven approaches.

## **2. The cognitive engineering approach**

The cognitive engineering approach will be presented using the example of ERATO, En-Route Air Traffic Organizer. This project is aimed at specifying and designing decision aids for En Route Air Traffic Controllers. It will result in a Controller's Electronic Assistant. The new philosophy of man machine cooperation central to its design will be described.

### **2.1 The steps of the project**

Eight steps can be identified in the process of designing cognitive tools. These steps are summarized in figure 1.

#### **2.1.1 Explication of the cognitive model**

Cognitive engineering has arisen from the expression of a need by numerous system designers : the need to understand what really makes the task difficult for the operators and how this difficulty impairs human performance, so as to define the most effective aids (Rasmussen 1986, De Montmollin & De Keyser 1985, Hollnagel 1988, etc).

We must not only elicit the knowledge of operators, but, first of all, we must understand how this knowledge is activated and utilized in the actual problem solving environment. The central question is not to identify the domain knowledge possessed by the practitioner, but rather to point out under which conditions this knowledge is (or is no longer) accessible. This is the problem of defining a validity domain for human performance. Cognitive engineering must also identify and predict the sources of error and the mechanisms of error.

When several agents can act on the system, under which conditions can they cooperate efficiently under time pressure ? What are the mental resources that are involved ? What is the cognitive cost of cooperation ?

## Méthodologie de spécification d'interface guidée par un modèle cognitif.

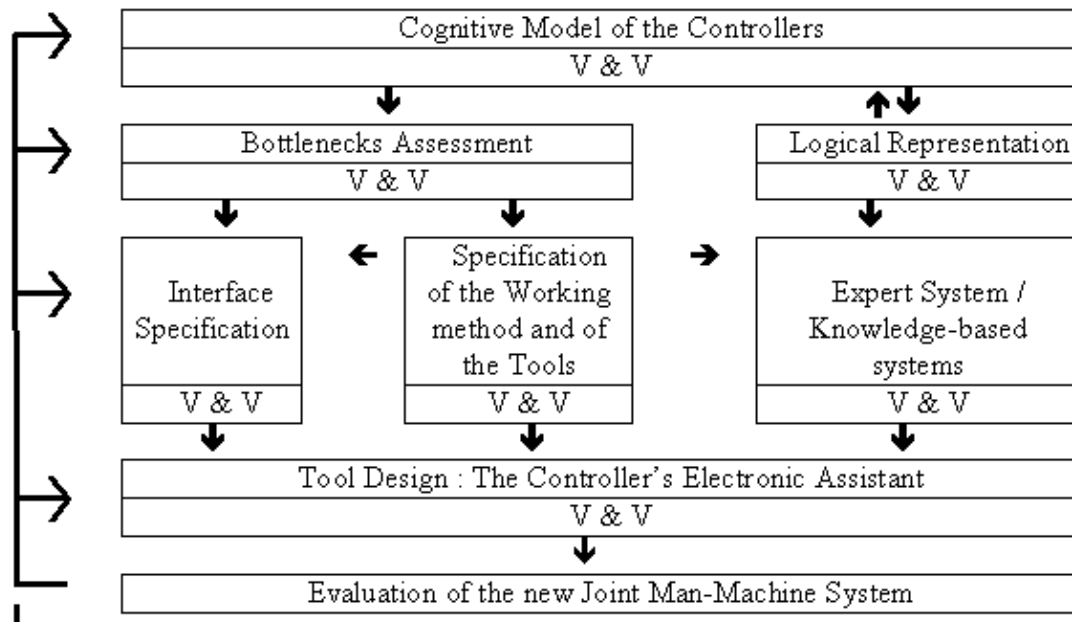


Figure 1. Eight phases for the design of the Electronic Assistant

It is necessary to point out in what ways the present tools are inadequate and how the operators compensate for the deficiencies of their tools. So it is necessary to examine how tools provided for an operator are really used by the operators.

All these analyses are needed to produce a satisfactory cognitive model of the operator. Such a model is central to defining a global approach to the design of effective decision aids.

This model identifies the mental mechanisms which are common to all controllers and which enable them to process data and to make real time decisions. These mental processes are analyzed for the executive controller, the planning controller and then for both controllers as a whole, so as to assess the consequences of cooperation on mental load as well as on global performance. The main goal remains to describe the mental mechanisms involved in the decision making process, and how these mechanisms evolve and decay under time pressure.

Four kinds of mental mechanisms or use of mental resources are described :

- those that are involved in the management of the physical process (i.e. maintaining sufficient separation between aircraft),
- those that are involved in cooperation between controllers working at the same control position,
- those that are involved in interface management,
- those that are involved in the management by the controller of his own cognitive resources.

The cognitive model is the key element of this approach, from the very specification of the functions up to the evaluation of the Joint Man–Machine system in operational conditions. So its formulation is critical. Several techniques have been used. But whatever the techniques, they must be ecological. The target must be the operator working in the multidimensional, open worlds which are his effective working context. The aim is to understand and describe the present–day mental activity of operators, given their present–day tools, and how these mental mechanisms decay under time pressure, fatigue and stress. When we cannot but experiment in laboratory conditions, the context must be as realistic as possible.

The first technique consisted in the observation of air traffic controllers in real working conditions, followed by interviews. A very creative technique consisted in combining both an Artificial Intelligence approach and a psychological approach, which enabled us to understand default reasoning and ambiguity elimination mechanisms.

### **2.1.2 Bottleneck assessment**

Decaying processes become bottlenecks in data processing and decision making. This step is a diagnosis phase: we have to point out the sources of poor and good performance of the air traffic controllers, given their actual working context. As long as the situation is not too demanding, controllers can compensate for these bottlenecks; but in very demanding situations these bottlenecks may severely impair the controllers' performance.

The initial explicitation of the cognitive model and the assessment of bottlenecks ended in 1991 in laboratory experiments. These experiments enabled the design team

- to verify that the main characteristics of the models were common to 10 controllers,
- to observe how these mental mechanisms evolve and decay under time pressure, fatigue and stress.

### **2.1.3 Functional specifications of decision aids**

The assessment of bottlenecks makes it possible to specify the basic functions of effective decision aids. Prior to this specification is the definition of the working method in an electronic environment :

- what do controllers actually need to build an effective mental representation of traffic in a cooperative way and under time pressure,
- how should the system support decision making processes.

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 man-machine system.

This distinction enables the organizing of the problems associated with the specification of man-machine interaction into a hierarchy. The man-machine interaction must meet the following conditions, cited according to their criticality :

- enable operators to exercise all the mental mechanisms that enable them to build the relevant mental representation of the system to be monitored,
- enable operators to cooperate in an efficient way,
- enable efficient inputs into the system.

These three points are necessary, but too often the third is the tree that hides the forest, while its single purpose is to ease the first two.

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 future tools is to pushback 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.

### **2.1.4 Interface specification**

The functional specification of decision aids implies the specification of algorithms, expert systems, and interfaces. The specification of these different components must be done as a whole. Interfaces cannot be specified independently of the algorithms, and vice versa. And the whole set of components cannot be specified independently of the definition of the working method : it is necessary to know in which context each tool will be operated, so as to optimize its 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 various inputs.

### **2.1.5 Definition of a logical representation of the model**

Some tools may require knowledge-based components. We need to combine different laboratory logics, semi-normal default logic, fuzzy logic and temporal logic, to build a logical tool adapted to formalize the controller's knowledge.

### **2.1.6 Encoding expert systems or knowledge-based systems**

The design of function defined during the third step may involve the use of knowledge-based systems which model large subsets of a controller's knowledge.

### **2.1.7 The design of cognitive tools**

These knowledge-based systems provide the cognitive tools with relevant data. So we have to face the problem of the integration of knowledge-based systems within real time software. This is one of the reasons that led up to the transformation of initial expert systems into purely algorithmic knowledge-based systems.

### **2.1.8 Evaluation, verification and validation of the joint man-machine system.**

Of course each of the previous steps include a local Verification and/or Validation phase. The feedbacks concern any of the previous steps, up to the cognitive model. This approach is necessarily iterative. The cognitive model is the spine of the project, that guarantees its convergence.

## **3. The case of air traffic control**

### **3.1 The cognitive model of the controller and the assessment of bottlenecks in decision making mechanisms.**

Those mechanisms can be described under four headings : mental mechanisms directly involved in the management of the physical process, (in our example, in detecting and resolving conflict situations between aircraft), the mental mechanisms involved in cooperation between different practitioners (in most countries two controllers are in charge of a control sector), the mental mechanisms involved in interface management, and lastly the mental mechanisms involved in own resources management.

#### **3.1.1 Mental mechanisms involved in the management of the physical process**

The following is a rapid overview of the controller's cognitive model as it is used in Erato.

### ***3.1.1.1 What really makes the task difficult***

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.

Controllers have to process data that depend on the time factor for :

- Their value
- 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 definitely unobtainable. Very often the controller must take decisions in a state of partial ignorance.
- Their accuracy. When observing air traffic controllers, we found that they spend a lot of time and a lot of cognitive resources, in eliminating ambiguity. A major reason why controllers are often unable to make a clear assessment of a situation is based on their representation of predicted time intervals : unless we sink into pure fatalism, we do not anticipate that an event will happen « at » a given time but « at about » a given time. The difference is fraught with consequences for the operator : decisions are always made in a fuzzy and uncertain context.
- Their flow. Controllers must adapt to sudden transitions from a situation of lack of data to a data overflow. Their mental and perceptive activities may be interrupted at any time by pilots' calls, telephone calls etc.
- For the time being, data presentation is technology driven and needlessly bulky.

### ***3.1.1.2 Making decisions in a state of partial ignorance :***

The controller anticipates according to a "normal", "routine" behavior of the aircraft, called the default behavior, with reference to the "default logic" that models this kind of reasoning. This default behavior is illustrated by controllers when they use sentences such as "normally this aircraft going to Paris Orly will start its descent about 30 NM before this fix". The controller does not know the top of the descent, but from his experience, he knows that "this will normally happen about here". So he will ignore all potential conflicts that might happen if the given aircraft should start descending earlier to focus all his activity on the most probable conflicts. This is an efficient means of narrowing the range of contingencies to be examined and to increase efficiency : at first all the aircraft are processed as if their behavior will always remain consonant with the "normal" behavior.

But to process exceptions, that is to guarantee safety, controllers monitor "sentry parameters". As long as these parameters remain in a "normal" range, all the previous diagnoses or decisions that are inferred from the default world, remain valid. But if a sentry parameter drifts outside the expected range, then all the previous plausible inferences have to be revised : some additional conflicts can be created, due to this abnormal behavior. In normal situations, this way of reasoning is an efficient and safe means to make decisions in a state of partial ignorance. But we can observe that, in very demanding situations, the monitoring task may no longer be performed by the controllers. Thus, when outside its validity domain, i.e. in too demanding situations, this mechanism may become a major source of errors.

### ***3.1.1.3 Making decisions in a fuzzy and uncertain environment***

At first diagnosis does not rely on the real world but on a mental representation of it. Even when remaining in the default world, the controller has to deal with a large set of data. Most of them are fuzzy, some must be actively acquired, and he must take into account the errors inherent in the system (including his ability to extrapolate). The controller is often unable to make a decisive assessment of the situation. Typically, ambiguity may arise when considering such questions as "will the separation between these two aircraft be

sufficient ?" or "what will the best maneuver to solve this problem be ?".

Allowing himself to doubt is a luxury for the controller. The mastery of doubt is an art. The controller faces the absolute need to eliminate ambiguity by pointing out and monitoring a few relevant parameters. To avoid a scattering of resources, these parameters will remain the only ones monitored. All other parameters are pushed into the background. For example, a basic heuristic approach in conflict resolution is : "*when two aircraft are converging towards a fix, the best solution is to vector the aircraft which will be the second to fly over the fix*". If at any time the controller is unable to decide which one will be the second on the fix, all his monitoring activity regarding this conflict will be centered on this point and he will ignore all other data concerning these aircraft.

This diagnosis mechanism is a means of organizing data functionally. It consists in a problem-driven and time-dependent organization of the raw data-set coupled with mental processes in order to refine strategies iteratively by monitoring the only relevant parameters. It significantly reduces the amount of data to be processed and it modifies their organization. For a sequential list of undifferentiated data is substituted a list of conflicts, certain or potential, with the main characteristics of these conflicts, their resolution-context and the relevant parameters enabling the controller to update this mental representation.

As a consequence the representation of the global situation is necessarily heterogenous regarding time. At any given time, the representation of the global situation is composed of several problems' representations, each of them triggering one or more resolution frames at different levels of abstraction. So the operator has to work simultaneously on different time scales. These frames result in the competition of data-acquisition tasks with different time spans. In addition to these tasks, new information from the system or requests from other operators may arise at any time and interrupt the current mental processes. The operator must shift from one task to another, from one time scale to another. This increases the cognitive cost of processing concomitant problems.

### ***3.1.1.4 Planning in an uncertain and highly dynamic context***

The previously described diagnosis-mechanism enables the controller to determine the requirements of the situation and to organize the raw data set. These requirements are immediately transformed into goals and subgoals by the controller.

In order to comply with a traffic requirement, the controller will have to act on the traffic, so as to modify at least one of the trajectories. The question for him is to decide which aircraft to act upon, when, and how. This is the problem of real time decision-making. As in conflict detection, the intention to act is not formed at one definite time but is the result of a process during which the mental representation of the problem will evolve from an abstract and schematic level to a level very close to the real world. During this process, the controller heuristically guides his activity towards the most promising directions by means of resolution frames.

The following is an example of a resolution frame for a typical problem, a radar vectoring between two converging aircraft (figure 2).



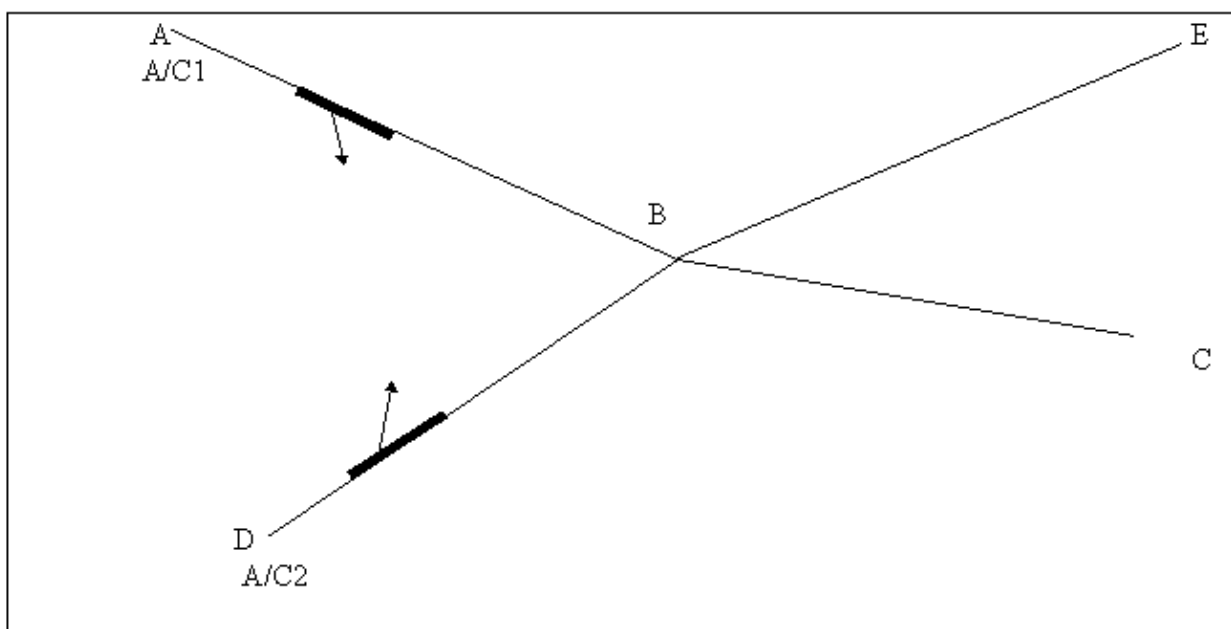


Figure 2

The controller knows that the first aircraft (A/C 1) will fly track ABC and that the second one (A/C 2) will follow track DBE. The controller may not yet see the aircraft on the radar display. The only information available comes from the paper strips. The conflict is certain or potential.

Considering only these prerequisites, any controller will state that

"The radar solution will be:

- either to vector A/C 1 to the right (arrow 1) and not to the left, or vector A/C 2 to the left (arrow 2) and not to the right. The choice depends on which one will be first over B (relevant parameter).
- This maneuver will be initiated when the selected aircraft is on the corresponding thick segment, (it is very important to note that the right time to act is not a time, but a position on the flightpath).
- The controller will have to monitor the position of the aircraft carefully, so as not to miss the correct space time span for delivering proper control instructions.
- He will then have to make sure that the pilot complies with these instructions
- He will then have to make sure that the maneuver is an effective resolution of the problem.
- Finally he will have to determine and monitor the end of the resolution in order to ask the pilot to resume his normal course."

#### 1. Frame triggering:

Resolution frames are triggered only by the statement of the problem. But the same problem can trigger several frames (for example a level change vs a radar vectoring). Most of the time the representation of the problem is initially ill-defined or inadequately defined. A part of the data-acquisition activity will be devoted to refining the representation, i.e. choosing between several triggered frames. In the previous example, if one of the two aircraft is climbing, the relevant parameter required for choosing a frame is the rate of climb. If the aircraft climbs very fast or very slowly, an intermediate-level solution may be chosen; if it climbs at a medium rate, a radar solution will be appropriate. As long as the controller has not clearly made up his mind, the tasks corresponding to the instantiation of the different triggered frames have to be performed in parallel.

As these different frames have different cognitive demands, the control and resources management mechanisms play a prominent role in this choice. Whereas the requirements of the situation are intrinsic to the default world, problem resolution depends on several personal factors. Isomorphic problems may not be processed in the same way by different subjects and even by the same subject on different occasions. A model of the controller cannot be deterministic. We shall now describe a whole frame-processing, from the instantiation up to the execution of the maneuver or up to its abandonment.

### 2. Frame instantiation:

The activity of the controller, regarding the previous example, will be centered initially on deciding which aircraft to act upon. The only relevant parameter is to determine which aircraft will fly over B first. The subgoal "decide which aircraft to act upon" involves a well specified data-acquisition task : "compare the evolution of both aircraft's positions until determining which one will be first". Then he will have to instantiate the maneuver schema, by defining more precisely the angle of deviation and the space-time span necessary to perform it. This is the end of the first phase, the decision making phase. The second phase of frame execution then begins, in which the controller delivers control instructions to the aircraft so as to modify its trajectory. This corresponds to a top down refinement strategy which is well adapted to routine problem processing. It enables the controller to point out the relevant pieces of information and to acquire and process these data very economically. During the first phase, data-acquisition from the real world is a means of refining and instantiating the resolution frame in an opportunistic way. At the end of this phase there is a profound conformity between the mental representation of the problem and the real problem. During the second phase, the controller checks that the real values observed or their extrapolations fall within the expected ranges.

### 3. Frame abandonment :

A frame can be given up for any one of several reasons:

- because an additional aircraft has changed the nature of the problem to be solved
- because of the effective values observed in the real world (e.g. the rate of climb has been modified, the level solution is no longer possible)
- because the controller missed the right space-time span for performing a decisive action
- because the pilot missed the maneuver, etc.

Of course, if a frame has to be given up, the controller has to shift to an alternative frame.

As a conclusion, a resolution frame is a schematic and abstract representation which allows the controller to guide his activity and to structure the data set efficiently. We indicate two levels in a resolution frame, these two levels being tightly interwoven : the goals level, which describes the intentional aspect of the activity, and the data-acquisition tasks level.

Frames are triggered by the problem statement : this implies that the aim of conflict detection is not only to indicate traffic requirements but also to produce an initial representation of the problem, so as to trigger the most promising resolution frames. Frames are refined, or abandoned from perceptual cues. To solve a problem, the controller can shift from one frame to another as necessary. This guarantees efficiency and flexibility in decision making.

At any given time, the representation of the global situation is composed of several problem representations, each of them triggering one or more resolution frames at different levels of abstraction. These frames result in the competition of data-acquisition tasks with different time spans. In addition to these tasks, requests from pilots or from neighboring executive controllers may arise at any time and interrupt the current mental processes. The controller must shift from one task to another. This involves very important memorization problems and emphasizes control and resources management aspects.

#### ***3.1.1.5 Memorization problem***

## Méthodologie de spécification d'interface guidée par un modèle cognitif.

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.

The controller must keep in mind:

- relevant traffic requirements so as to be sure that his current activity complies with them.
- the decisions to act,
- the triggered frames,
- the active goals,
- the associated data–acquisition tasks, as well as the targetted time spans to perform these tasks and their logical consequences. While data organization allows very efficient processing of information, it also furthers chunking. Early studies on controllers' memory (Sperandio 1969, Sperandio 1974) show that experienced controllers are able to keep in mind a greater amount of data than beginners. Data relevant to conflict resolution are more easily memorized. We observed three major problems regarding memory.

### 1– Temporal deadline memorization:

Controllers are very sensitive concerning the "right time" to perform actions on traffic, (even if the right time is in fact the right place to act). It is easy to observe how an overly tardy action on traffic may transform a fairly difficult problem into a very difficult one. Controllers are unable to memorize time spans. To make sure that they do not miss the right time for delivering a control instruction, they very frequently monitor the exact position of the aircraft, until it enters the targeted area. Obviously this mechanism is very costly. When a maneuver has to be performed in a very acute time interval, monitoring the "right time" to act becomes a demanding task. The controller must shift frequently from one problem to another. At each shift he has to restore the resolution context. When conflicts are complex and subject to time pressure, this may become a critical task.

This is very expensive in terms of cognitive resources, for it requires very frequent shifts from other pending problems. A subsidiary role of monitoring the system is to synchronize both the system's and the operator's internal "clocks".

### 2– Frequent goal–shifting and context–restoration:

When controllers are confronted with a few simple and routine problems, chunking allows them to keep all the relevant information in mind. But when the situation becomes more demanding, when problems become more complex, controllers have to restore the context at each shift. Some problematic aircraft may be hidden amongst clusters of problem–free aircraft. Even for experienced controllers, recovering all these data under time pressure may become difficult.

The efficiency of the mental processes also depends on the capability of the operator to focus his attention on the relevant problem at the right time, and to control the focus of attention, so as to avoid tunnel–vision effects.

### 3– Memory fading with time:

The controller has to save sufficient cognitive resources to refresh all the previously stored data periodically.

Triggering the adequate knowledge into the working memory under time pressure has a cognitive cost that must be taken into account.

### 3.1.2 Cooperation

Most of the previous tasks can be performed by the two controllers, successively or in parallel. Mental mechanisms involved in cooperation are an essential part of the model. Efficient cooperation 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.

When demand increases, these latter two conditions may decay so much that cooperation may no longer be effective. Numerous airmisses have been reported that are due to cooperation failure in too demanding situations. One controller did not even know that some tasks were urgent and important while the other controller thought that these tasks had been normally performed. This points out the limits of cooperation based on implicit task delegation and implicit task allocation.

### 3.1.3 Interface management

The interface can be thought of as a window, through which the operator can get only a partial and distorted view of the real world. Mental mechanisms enable the controller to build the effective mental representation, i.e. the representation that enables action on the world to be taken.

The present data displays (strips and radar) are not well adapted to the controllers' mental activity. The executive controller organizes the strip rack according to the potential conflicts that he will have to solve. But we can observe that this is not a reliable means to exchange data between controllers : in demanding situations, when cooperation should be the most efficient, it becomes impossible to update the organization of the strips.

This does not guarantee that both controllers have a relevant representation of the effective traffic requirements. This is the reason why a problem-driven information display seems to be necessary. This kind of tool will enhance the efficiency of the mental mechanisms and will improve cooperation between controllers without constraining their activity. It will be one of the main components of the electronic assistant.

### 3.1.4 Cognitive resources management

The controllers' activity as described above consists

- in the creation of a representation of how the world will evolve over time, associated with parameter-monitoring processes to guarantee the soundness of this representation,
- in potential problem assessment so as to determine the requirements of the situation,
- in real-time-decision processes,
- in the monitoring of multiple relevant parameters so as to meet different goals during different time spans.

To all these monitoring processes can be associated

- a time range with loose boundaries or with a mandatory deadline; the boundaries as well as the deadlines are often conditional ones,
- a tempo which may vary over time according to the internal logic of the task or due to the consequences of cognitive resources management. This is the reason why (i) the representation of the

## Méthodologie de spécification d'interface guidée par un modèle cognitif.

world does not evolve continuously over time but in a discrete way, (ii) the representations of each problem do not evolve at the same rate.

All these processes are highly interactive. In demanding situations they may severely compete with and constrain each other. These multiple mental processes are often interrupted by events from the real world (pilots' requests, neighboring executive controllers' requests). The controller has to attribute limited cognitive resources to these multiple relevant goals. During the validation 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 amongst several pending ones and to perform it in the right time span. Resources management depends, amongst 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.

Resolution frames give an example of how operators plan their activity. As the operator has to divide his cognitive resources among all pending tasks, resources management mechanisms will strongly interfere with planning and control mechanisms. We can extend the previous notions to cognitive resources management. Without trying to establish an exhaustive taxonomy of such strategies, significant changes over time in Air Traffic Controllers' cognitive-resources management can be observed easily. These changes may occur very suddenly. They mostly depend on the recent, present and expected taskload assessment. These changes do not only concern the tempo of tasks completion but also significantly modify the current planning strategy as well as the resolution-frame policies or tactics. The cognitive cost of a resolution frame depends on these factors.

Resolution frames give an example of how controllers plan their activity. As the controller has to divide his cognitive resources among all pending tasks, resources management mechanisms will strongly interfere with control mechanisms.

Resources management depends :

- on the short term situation assessment: the number of tasks to be performed with their time span, their status, their technical difficulty and their coupling. The goal-switching difficulty depends on the number of shifts and on the number and complexity of each context restoration,
- on the controller's representation of the working context (the neighbouring executive controllers, the planning controller), which raises the problem of trust among operators.
- on the controller's representation of his own capabilities.

Depending on these factors, the executive controller divides his cognitive resources accordingly. For example "I feel short of time, solve the urgent conflicts first, postpone conflict detection" or "expedite the resolution of conflict A so as to have time to plan the resolution of conflict B". This division gives rise to a philosophy that is used to plan and control each problem resolution. This philosophy is "be efficient first" for problem A whereas it is "be elegant first" for problem B.

Let's consider the situation shown in figure 2, with A/C1 and A/C2 converging on fix B, and let's assume that A/C2 is flying at 29000 ft (ie flight level 290), while A/C1 is taking off and wants to climb to 35000ft (ie flight level 350).

Figure 3 shows the consequences of different cognitive resources-management strategies at the goals level of the resolution frames.

<b>Be elegant first</b>
<p>I climb A/C1 initially to level 280 (1) Then monitor its rate of climb (2)</p> <p><b>If</b> it climbs slowly, a little after B it will climb to level 350</p> <p><b>If</b> it climbs very fast, I climb it to level 350 and monitor carefully (3)</p> <p><b>Otherwise</b> : I provide radar vectoring and climb it to level 350</p>
<b>Be efficient first</b>
<p>I climb A/C1 initially to level 280 (1)</p> <p>I provide radar vectoring and climb it to level 350 (4)</p>
<b>Safety and nothing more</b>
<p>I instruct A/C1 to climb to level 280 and maintain (5)</p>

Figure 3

**Comments :**

1. Level 280 is called "the safety level"
2. Monitoring a rate of climb is very demanding, as uncertainty on this parameter is very high. Its value may change suddenly. To avoid risk, the controller has to focus on this aircraft very often, which increases the number of shifts from one problem to another, and increases the cost of dynamic memorization processes.
3. As the resolution depends on an unreliable parameter, the controller has to monitor very carefully so as to take the appropriate decisions if necessary.
4. A radar vectoring, especially in a routine situation, is not very demanding. Even if it turns out to be unnecessary, the controller prefers this solution, which guarantees both safety and efficiency, rather than spending a lot of time eliminating ambiguity. This clearly shows that the results of conflict detection mainly depend on cognitive resources management strategies !
5. This solution occurs in very demanding situations. As A/C1 remains below A/C2, the conflict is suppressed. This solution is the most economical for the controller but may be penalizing for the aircraft.

Different philosophies also have noticeable consequences at the task-level of the frames. Data-acquisition tasks are organized into a hierarchy. The data-acquisition tempo may change significantly. Some tasks may be postponed or cancelled. For example, in very demanding situations, the relevant

sentry-parameter-monitoring tasks may be performed as background tasks. The executive controller may delegate some tasks either to pilots (rather than monitoring a parameter, the controller asks the pilot to do this: "report when...") or to the planning controller. In this domain, the most important mechanism is the implicit task-delegation. According to his representation of the planning controller, i.e. his confidence in the other operator, the executive controller may disregard less urgent or less important tasks. The planning controller will detect these tasks, monitor them and advise the executive controller when a specific action is needed. Cooperation is also a function of cognitive resources management strategies.

To summarize, a part of the activity of the controller is devoted to choosing the best frame. Each of these frames may be more or less demanding. In demanding situations the cognitive cost becomes a basic criterion in choosing a frame. Of course while resolving a problem, the controller may have to shift from an inoperative frame to a more relevant one.

According to the assessment of his workload, the controller can instantiate a resolution frame in a more or less efficient way. He can also abandon a more elegant frame to shift to a more efficient one: this is the consequence of his own resource management policy, according to the problems he has to face at any given time.

All these mechanisms are a part of the real time data process. This process results in a problem driven organization of the raw data set which enables large amounts of data to be processed.

Mental mechanisms have a validity domain. It is easy to observe how their efficiency decays under stress, time pressure or fatigue. For example, in demanding situations, the sentry parameters are less monitored and this may lead to errors when abnormal behavior is not detected soon enough. The assessment of a given conflict, including conflict detection and resolution assessment, needs a few seconds when the number of aircraft is low, while it can take 7 to 8 minutes in very demanding situations ; in this case, the controller is confronted with problems associated with numerous shifts from this conflict to concomitant ones, as described above and the risk of error (forgetting a relevant aircraft, choosing the wrong resolution frame, etc.) is high. The validity domain of the mental processes directly depends on the number of aircraft that have to be processed by the controller. This is the reason why we focused on a problem driven presentation of the information.

### **3.2 The functional specification of decision aids**

#### **3.2.1 Guidelines**

For the next decade, the nature of the data to be processed by the controllers will not significantly change, only their volume. So, all the mechanisms that are inherent in the nature of data to be processed must be preserved in the new environment.

But 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 economize cognitive resources),
- or to manage them in a more efficient way.

The justification of the tools is a key point of the approach. It explains the reasons behind the design of each function, and the improvements of the joint Man-Machine system that are expected. It also defines the criteria

used in testing this system. At this level, there must be a profound symbiosis between theoretical work on the cognitive model, design and validation.

This implies that the design team must be multidisciplinary and multilevel, i.e. include researchers and practitioners in each discipline.

### **3.2.2 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.

#### ***3.2.2.1 Task driven data presentation***

For any situation we can identify the very few data that are useful for a given task, the electronic environment 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.

For example, to manage the flows of outbound traffic, the planning controller does not need all the data that are shown on the present flight progress strips. He only needs 5 or 6 items, compared to the more than twenty that are available now. The flight progress strips are organized on a rack according to the needs of the executive controller. This organization does not meet the real needs of the planning controller. The electronic interface makes it possible to show the planning controller the exact subset of data that he needs, organized in a way that provides an efficient support to his cognitive needs.

#### ***3.2.2.2 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 have be able to process these subsets of aircraft very efficiently. As we do not intend to provide the controllers with the results of automatic conflict detection and resolution, they will have to use all their mental mechanisms to assess the situation. This should preserve their skills and their capability to deal efficiently with any unanticipated situation. The expected gain is that, as they will be working on appropriate subsets of aircraft, these mental mechanisms will be much more efficient than now.

Thus, we have to verify that this man-machine cooperation philosophy enhances all those mental mechanisms that are inherent in the very nature of the data to be processed, as identified in the cognitive model :

- The way controllers anticipate in a state of partial ignorance.
- The associated sentry parameter monitoring processes.
- The ambiguity elimination processes : the definite assessment of the situation should be made earlier and in a more acute way than now.
- The choice of the relevant resolution frame should be made earlier than now, and in a more « elegant » way.
- The cooperation between controllers should be improved. The information filtering is supposed to enhance the definition of the mental representation of traffic. Both controllers' mental representations of the situation should remain consistent over time, as they will be able to update them very easily.

#### ***3.2.2.3 Meeting cognitive needs as closely as possible with the interface***

The following is an example of how the cognitive model is used to specify the interface in Erato.

It is commonly admitted that operators spend a significant part of their activity in compensating for tool



deficiencies. An ill-adapted interface can significantly devalue the results of information filtering. The extrapolation function of Erato allows one to substitute a graphical representation for an alphanumeric one. Experiments have shown that most of the time the referential used by the controller is not a temporal one but a spatial one : the question "When will you act on this aircraft ?" is answered "There". So the interface will enable the controller to drag an aircraft along its trajectory with the mouse ; all the other aircraft will move accordingly. This interface really meets the way the controller anticipates. If this interface had had a temporal referential, the controller would have had to mentally convert distances into time intervals ; in demanding situations this could represent a significant additional workload.

### **3.2.3 Improving the management of cognitive resources**

The problem-driven information filtering allows the controller to focus all his activity on well-formulated problems ; so he can operate all his mental mechanisms in a more efficient and creative way. This function substitutes a set of easily manageable problems for the initial complex situation.

The 1991 experiment has shown how difficult it may become for the controller to focus entirely on the right problem at the right time. The solution proposed in Erato is a new function, the reminder. The reminder consists of a specific window of the electronic assistant 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 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.

The aim of the reminder is to show the two controllers what the traffic requirements are and their urgency. Thus it should enhance cooperation between them. There are several ways to split a given situation into relevant problems. This variability can be observed for several different controllers as well as for any given controller, according to his cognitive resources management philosophy. The more demanding the situation is felt to be, the more the controller will split it into « little » problems and solve these problems in a very tactical way, with short term solutions. If the controller feels that the situation is mastered, he will consider these elementary problems as parts of a whole and solve them in a more strategic way. Thus, 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 reminder should be used by both controllers as a safety net based on intentions of action.

## **3.3 The role of expert systems or knowledge-based systems**

### **3.3.1 Expert systems : solving the problem vs formulating the problem properly**

The role of expert systems or knowledge-based systems as defined in Erato is not to solve the problem (detect conflicts and/or solve them). The problem-driven information filtering allows the controller to focus all his activity on well-formulated problems ; so he can operate all his mental mechanisms in a more efficient and creative way. This function substitutes a set of easily manageable problems for the initial complex situation. The role of expert systems is "just" to assist the controller in formulating the problem in a more efficient way. Operators no longer feel they are being progressively expelled from the decision making loop, but that they are more powerful thanks to the machine.

### **3.3.2 The logical representation**

As mentioned previously, two different logics need to be combined to formalize the different facets of the controller's reasoning patterns, a seminormal default logic to model how controllers anticipate in a state of partial ignorance, i.e. from the "normal" behavior of any aircraft, and a fuzzy temporal logic to represent ambiguity elimination processes.

### **3.3.3 The knowledge-based system and the main information filtering function**

The role allotted to the knowledge-based system is to provide these electronic assistants with adequate data, to show

- how to organize the raw data in a problem-driven way
- what the traffic requirements are and their urgency.

#### ***3.3.3.1 Description of the knowledge-based system***

The first version of the expert system included about 3,000 Prolog first order rules. This expert system is no longer used by the system. Algorithms have been derived from the expert system and are encoded in C++.

These algorithms process the same set of data as the controllers have to process now, i.e. the information from the flight progress strips and, when available, the radar information.

The knowledge-based system includes two main modules. The first one computes the default representation of each aircraft. From this representation, the second module associates with each aircraft its relevant environment, called Interfering Aircraft Subset (IAS).

This environment is composed of :

- the subset of all conflicting aircraft. These conflicts may be certain or potential. This subset is not determined by means of a pure mathematical computation, but according to current expertise of controllers.
- the subset of all the aircraft that may interfere with a "normal" radar resolution of the conflict, that is all aircraft that may constrain conflict resolution. A normal resolution is a solution which is consistent with the current know-how of controllers.

The relevant environment of an aircraft is typically a problem-driven filtering of information. The IAS represents the relevant working context associated with an aircraft. Such an environment embodies traffic requirements and all information that may be useful to fulfill these requirements. The number of rules is explained by the need to represent the skills and knowledge possessed by controllers to day so as to make sure that information filtering really meets the controller's needs.

The definition of the relevant environment is given by the following :

"according to the traffic requirements, provided the EC works normally, he may need all, or a part of, the displayed data, but he will in no way need any other data".

Information filtering techniques are under dispute (De Keyser 1987). The point is how to make sure that the operator won't need a piece of data that is hidden by the system ? Such data retention would be an unacceptable source of errors.

#### ***3.3.3.2 Discussion***

The discussion on the exhaustiveness and the relevance of data filtered by the machine is central. The basic answer consists in providing the operator with functions that enable him/her to access extended sets of data, or even the whole set of data. This solution is not relevant when using a problem-driven filtering system.

- The first answer consists in taking into account the default behavior of the aircraft in a more "prudent" way than the controller. This will result in the display of some aircraft that may be not relevant for the controller. In the most demanding situations encountered by controllers now (more than 30 aircraft in the sector), most IAS include less than 8 aircraft. If one or two additional aircraft are displayed this is not really a problem : in all cases, the number of aircraft displayed as a result of information filtering remains lower than the maximum efficient processing-capability (about 15 aircraft), while the initial number was significantly above this figure.
- Then the system detects all potentially abnormal behavior of an aircraft, so as to advise the controller as soon as possible and to update information filtering accordingly. In future versions, this mechanism should be performed using FMS/Data-Link capabilities.
- But these first two answers do not really solve the problem. The knowledge elicited in the expert system defines a set of "normal behaviors" on the part of the controllers. But whatever the number of rules can be, it is impossible to represent the total knowledge of the controllers. To be able to do this, we should have to deal with controllers' errors or creativity. The solution defined in Erato consists in considering the The knowledge-based system as a default representation of the controllers. To guard against the consequences of human error or creativity, (ie unexpected behavior) a monitoring process is associated ; this process is inspired by the natural sentry-parameter monitoring process of the controllers. This monitoring process will detect any discrepancy between the actual position of all aircraft and any of the possible positions that could result from a "normal" behavior of the controller. When necessary this process will trigger an alarm, so as to advise the controller that the previous information filtering is no longer relevant and has been updated. We have to make sure that this mechanism is efficient in demanding situations, and that the controller is not interrupted by the warnings too often. In other words, the The knowledge-based system must be sufficiently accurate.

This monitoring process linked to the knowledge-based system allows the Electronic Assistant to adapt very smoothly to operator error and creativity. Such information filtering is error tolerant.

### 3.3.4 Validation of the problem-driven filtering algorithms

This validation was carried out in 1993, using five very demanding traffic samples from five different upper sectors. These experiments involved 44 FPL controllers from the five French control centers.

The protocol consisted in interviewing controllers at identified moments of the simulations, to ask them the following questions on target aircraft (about 25 per traffic sample) :

For each aircraft,

- describe what its behaviour will be inside the sector ,
- describe all the conflicts (certain or possible),
- for each of these conflicts, describe the resolution frames that can be used
- for each of these resolution frames, describe the aircraft that are, or may be, a constraint.

Then, the results were merged and compared to the results of the algorithms.

The first result is that the algorithms are generic to all French upper sectors, and needed very minor adjustments.

The second result is that the rate of filtering is correct. Despite the fact that the number of aircraft in the sector was between 30 and 40, (about twice the present peak of demand), on 122 target aircraft, there were only 4 IAS higher than 15 aircraft (15 aircraft is considered by controllers as their maximum efficient processing-capability), while 92 IAS were less than 8 aircraft.

### **3.3.5 How the controller's electronic assistant uses the outputs of information filtering**

The basic information filtering will be used by the following functions:

- Simulation functions

These functions will allow the controller to test different resolution frames and to answer questions such as "What would happen if I climbed this aircraft to ..." or "Is it better to set course directly to ...". The expert system will deliver the simulated information filtering. These answers will be updated until the controller has made his choice. For the time being the controllers have no tool that assists them to perform this task.

- Memorization aids

The controller will have the capability to indicate the trajectory section where he intends to act on an aircraft. When the aircraft flies over this point (or abeam of it), a warning will be triggered. Then, after having consulted the filtered information, the controller will instantiate his decision. This should solve both problems of keeping in mind the "right time to act" and real-time context-resolution updating.

## **3.4 Evaluation, verification and validation of the Joint Man-Machine System**

### **3.4.1 Methodology**

Classically we define dependability as that property of a computing system which allows reliance to be justifiably placed on the service it delivers (Laprie 1987). We can point out four classical methods regarding dependability—procurement or dependability—validation : fault avoidance, fault tolerance, fault removal and fault forecasting. These definitions can be applied to a complex heterogeneous man-machine system such as the ATC system, as well as to any of its machine components. In the first case the users are the airlines (or their passengers), while in the second case the user is defined as the controller or any subsystem.

The specification of decision aids relies on a philosophy of future man-machine cooperation, whether this philosophy is clearly defined or not. The central question is to make sure that this cooperation fulfills the initial requirements regarding capacity and safety. To answer this question, we have to choose the right parameters to be evaluated, then we must determine the minimum set of experiments to get a significant amount of data (Woods and Sarter 1992). Some basic questions must be answered, about the robustness of the joint man machine system : is it error tolerant (Foster 1992) ? Does its organization allow a quick and easy correction of errors ? (Reason 1992).

The design of decision aids implies an analysis, either implicit or explicit, of operator deficiencies and of the most effective means to compensate for these deficiencies. The ultimate step of the verification/validation process should be the verification of these initial assumptions (Hopkin 1992).

The first level of validation consists in testing both the working method and the interface so as to verify that it really improves the target bottleneck and to make sure that it does not adversely affect some sources of good performance.

## Méthodologie de spécification d'interface guidée par un modèle cognitif.

Then we will have to assess the validity domain of each function : is it really efficient in situations where the controller needs an effective aid ?

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

- How well is this cooperation philosophy accepted by controllers ?
- How will it modify their activity ?
- Does it enable them to work in a more efficient and creative way ?
- Does it provoke a loss of vigilance or of skill ?
- Does it improve the global performance, from the capacity and safety points of views ?
- Does it enable a progressive and "soft" integration of technological advances in avionics ?
- What are the consequences for training ?

### **3.4.2 The first evaluation campaign October 1994–April 1995**

#### ***3.4.2.1 The role of the cognitive model in defining the experimental protocols***

The cognitive model provides a guideline for evaluating the joint man-machine system. It enables the transformation of high level validation requirements into relevant criteria to test the joint man-machine system. 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. Then one can determine or assess the gains in these aspects.

We must verify that the new joint man-machine system preserves the sources of good performance and really improves the weak points from both a safety and a capacity point of view. This suggests that we must assess the performance of the new system with reference to the previous one in real conditions, i.e. whatever the variability of the real world is, in demanding and very demanding situations. The experiments should enable one to determine how the joint man machine system 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, and what are the consequences for operator training.

The experimental protocol was based on the comparison of the performance of controllers in demanding or very demanding situations, in the present environment and in the future one.

The experiments involved nine teams of two controllers. Each team of two controllers was confronted with four traffic simulations, two of them using a conventional environment (paper strips and a radar display), the two others using the Electronic Assistant. The average duration of a simulation was 45 mn. A video record of the controllers' activity and a computer record of all the actions on the system was made. Then controllers had to fill in a questionnaire on the functions : for each function they had to answer questions on the relevance of the function (regarding safety, capacity, cooperative activity, etc), its usability, the function as a (potential) source of error, their opinions on planned improvements of each function. Lastly, they had to comment on their own activity in a semi directed way, using both the video replay and the computer replay of the simulation. The interview was orientated on the workload as it was felt by both controllers, and focused on the consequences for conflict detection, conflict resolution and on cooperation, so as to get data on resource management and to compare with equivalent situations in a conventional environment. The average duration of the interviews was about five hours for each traffic simulation.

#### ***3.4.2.2 The role of the cognitive model in training***

For operational reasons controllers were available for two weeks only. This was a serious constraint, because they had only 6 days to train on ERATO before the evaluation tests. To optimize 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 cooperation.

Obviously, the use of a mouse is a problem under real time conditions. But the most difficult task for controllers consists in acquiring new skills to build a relevant representation of the real world through the new interface : what are the relevant cues ? where to pick them ? how to interpret them ? All the cues they are used to, and all the associated reflexes, depend on the present interface : they are no longer available. Controllers must learn how to restore an efficient visual circuit on the interface, i.e. acquire the ability to make efficient use of all the visual cues provided by the new system.

In the same way, present cooperative 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 cooperation in an electronic environment, how to make the tools improve this multimodal activity rather than sterilizing it.

Only the first day was dedicated to learning the logic of the machine and the basic handling of the interface. For five days, training was directed towards the acquisition of the apomorphic skills, always with reference to the present ones : « The goal remains the same : building a robust mental representation of traffic, the way of doing it is different »

### **3.4.3 Results**

#### ***3.4.3.1 Positive results***

As shown in figure 1, the experiments have feedbacks at all the levels of the project. Whenever a function is not satisfactory, it must be assessed whether it comes from the working method, the interface design, the algorithms, the specification itself, or from bottleneck assessment. Whatever the reason is, we have to refine the cognitive model. Different opinions of controllers in the questionnaire were set against the present state of the cognitive model so as to improve it and to understand the genuine reasons behind those diverging opinions.

A statistical analysis of actions on the interface, with reference to the number of aircraft, of problems etc..., is cross checked with the comments so as to analyse how each function is really used over time when the demand varies. Combined with the analysis of the video record and of the interviews, this statistical analysis enables us to understand how new tools will modify controller activity, to verify if the decision aids enable them to work in a more efficient and creative way without provoking a loss of vigilance or of skill, and to make sure that they improve the global performance of the joint man-machine system, from capacity and safety points of view.

The video analysis is not yet completed. However, initial results, based on debriefings, on interviews, on questionnaires and on statistical analysis show that this automation philosophy is very well accepted by 15 controllers, rejected by one, and that two controllers asked for additional tests. The controllers think that these decision aids will considerably decrease their stress, and thus will enable them to increase capacity and safety significantly. We can already consider that the operational concept is validated. Nevertheless a lot remains to be done to improve the usability of all the functions. Some algorithms of the reminder have also to be modified. These experiments are the initial phase of an iterative process leading to final specification of the Electronic Assistant by mid-1997. Final experiments with this interface connected to the second version of the algorithms are planned by the end of 1997, so as to specify the first operational version by mid 1998.

### **3.4.3.2 *The reminder***

The reminder is the main factor in increasing productivity. But its efficiency relies on efficient cooperation between controllers. Three agents can act on the reminder : the algorithms and both controllers. The experiments showed that the role of each agent needed to be clarified. The refining of the reminder have been an exemplary illustration of the methodology used by the team.

The starting point was the confidence that the reminder was central

- in building the mental representation of traffic
- as a support to cooperation.

From these initial assumptions we worked with controllers so as to determine what the most efficient working method with the reminder should be. From that working method, we inferred the specification of the second version of the interface of the reminder. A mock up was designed and connected to the whole interface. This enabled us to test the working method and the interface under realistic conditions, and to refine both.

### **3.4.3.3 *The cues***

The electronic environment deprives the controller of all relevant cues that were familiar to him in the previous environment. This is not only a matter of nostalgia. 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,
- the controller can easily detect and process them.

This task is central to the interface specification. It is made more complex in an electronic environment because of the saturation of the visual channel. Thus we have to define a global policy of data presentation on the interface, and in fine, a working method relying on relevant visual circuits on the interface. This definition process will make use of more advanced studies on the attention and the focus of attention.

## **3.4.4 Integrating the results**

### **3.4.4.1 *The cognitive model and rapid prototyping***

The definition of the Joint Man–Machine System is necessarily an iterative process. From the cognitive model, we can infer functions, sketch out working methods and the associated interfaces. Experiments under realistic conditions make it possible to refine all of these. Then rapid prototyping techniques can be used to analyse working methods step by step. Controllers are confronted with simulated traffic samples, so as to analyse what the actual cognitive needs are for the building of an effective mental representation of the situation, and how the machine should assist the operator in this task. The main object of rapid prototyping is not the interface, but the working method. The interface is specified only when the working method is clearly defined.

### **3.4.4.2 *The role of users in a multidisciplinary team***

It is often difficult to accurately define the role of future users when specifying and evaluating a new working environment. The cognitive engineering approach makes it possible to solve this problem in an efficient way, thanks to the reference to the cognitive model. About 25 Air Traffic Controllers attend working sessions with

psychologists and/or AI specialists and/or computer engineers, in teams of two or three.

The behavior and the opinions of each controller is analysed according to the cognitive model. In the first place this makes it possible to enrich it. Thanks to the cognitive model, the variability between people can be taken into account, not to find an unreachable consensus between opposite opinions, but to understand what different (and not directly formulated) cognitive needs lay behind those different opinions, so as to translate these cognitive needs into working methods and an interface specification.

The specification of the tools is not directly driven by future users. The cognitive model is an irreplaceable level of abstraction between the users and the designers.

### ***3.4.4.3 Cognitive engineering is ecological***

The cognitive model must be continuously amended, from observations of the real working context. Laboratory experiments, realistic though they can be, are not sufficient. The final experiment, before the operational deployment, will be held in operational conditions by mid 1999. Real traffic will be controlled, using the Erato environment, in a control sector protected by a conventional shadow sector. A lot is expected to be learnt by observing how people really use the functions. The analysis of the discrepancies between what was expected and what really happens will be a major source of improvement for the cognitive model.

### ***3.4.4.4 How to manage with "external" technological advances?***

The cognitive model is about the present system, including present operators, etc. The evolution of the Air Traffic Control system must include major technological advances such as Data-Link, precise trajectories computed by onboard computers (FMS), advanced navigation means enabling very long direct tracks. The technical features are now well-known, but very little is known about the consequences on the Joint Man-Machine System.

One can easily imagine an ideal system within two or three decades, from a purely technological point of view, but what about the transition from the present system to the future one ? Cognitive Engineering seems to be a powerful means of mastering the evolutions of the system, as the cognitive model makes it possible to focus on relevant problems that will necessarily be met by practitioners in such an environment, so as to foster studies in critical areas.

For example an intensive use of Data-Link will put an end to the executive controller's monopoly of interaction with pilots. Thus all the cooperation mechanisms between controllers will be modified. The elicitation of the present cooperation mechanisms, and how these mechanisms contribute to the building of the mental representation of traffic, makes it possible to anticipate how they will be modified, and to do experiments in this domain.

Thanks to Data-Link, a very accurate trajectory may be down linked to the ground. In consequence the very nature of data that must be processed by controllers will change. This will involve tremendous changes in cognitive activity, in default reasoning mechanisms, in ambiguity elimination processes, in plausible inference, in data-acquisition mechanisms, etc. This will be the case for those aircraft that are equipped, but not for all aircraft. In such a turbulent transition, the cognitive model provides the guiding lines to the tools' designers, so as to master the different facets of the specification and evaluation of the future joint man-machine system.

## **4. Conclusion**



## Méthodologie de spécification d'interface guidée par un modèle cognitif.

The cognitive engineering approach applied to Air Traffic Control proves to be successful. The backbone of this approach is the cognitive model of Air Traffic Controllers. The activity of the design team is driven by this model, either to infer the specification of the tools, or to search for relevant criteria to evaluate the new Joint Man–Machine System, or to improve the cognitive model itself.

As a consequence the design team must be multidisciplinary, including psychologists, ergonomists, AI specialists, HMI specialists, computer engineers, Air Traffic Controllers. It must also be multilevel, including in each subject researchers and practitioners. There must be a continuous interaction between concrete problems and fundamental research inside the team.

The approach must continue after the operational deployment so as to analyse how people really use the tools, i.e. how the new Joint Man–Machine System really works, so as to identify and interpret the differences between what was expected and what really happens. This phase is very important, both to enhance the cognitive model and to improve the design of the tools.

Identifying at the outset what is critical in the operators' activity makes it possible for the designer to focus on relevant questions during any validation phase.

We have shown how this principle driven design results in a far more efficient use of training, which is not thought about as a means of compensating for design inadequacies, but as the continuation of the design process. Both the cognitive model and the choices made during the design process provide guidelines for training. Identifying apomorphic and plesiomorphic features in the new Joint Man Machine system is of great interest to make training more efficient.

This approach has been in place for a long time, about 15 years between the very first studies and the deployment of the operational version. But this duration is comparable to other major projects in ATC. At the beginning of the process, it seems costly, as the first mockups are a long time coming, but in the final analysis, the cost/benefit ratio is very positive.