Handbook of Human Factors & Ergonomics, second edition, G. Salvendy (Ed.), Wiley, 1997

AUTOMATION SURPRISES

N.B. Sarter, D. D. Woods, and C.E. Billings
Cognitive Systems Engineering Laboratory
The Ohio State University

The road to technology-centered systems is paved with user-centered intentions.
(Woods, 1994)

1 INTRODUCTION

In a variety of domains, the development and introduction of automated systems has been successful in terms of improving the precision and economy of operations. At the same time, however, a considerable number of unanticipated problems and failures have been observed. These new and sometimes serious problems are related for the most part to breakdowns in the interaction between human operators and automated systems. It is sometimes difficult for the human operator to track the activities of their automated partners. The result can be situations where the operator is surprised by the behavior of the automation asking questions like, what is it doing now, why did it do that, or what is it going to do next (Wiener, 1989). Thus, automation has created surprises for practitioners who are confronted with unpredictable and difficult to understand system behavior in the context of ongoing operations. The introduction of new automation has also produced surprises for system designers/purchasers who experience unexpected consequences because their automate systems failed to work as team players.

This chapter describes the nature of unanticipated difficulties with automation and explains them in terms of myths, false hopes, and misguided intentions associated with modern technology. Principles and benefits of a human-centered rather than technology-centered approach to the design of automated systems are explained. The chapter points out the need to design cooperative teams of human and machine agents in the context of future operational environments.

Automation technology was originally developed in hope of increasing the precision and economy of operations while, at the same time, reducing operator workload and training requirements. It was considered possible to create an autonomous system that required little if any human involvement and therefore reduced or eliminated the opportunity for human error. The assumption was that new automation can be substituted for human action without any larger impact on the system in which that action or task occurs, except on output. This view is predicated on the notion that a complex system is decomposable into a set of essentially independent tasks. Thus, automated systems could be designed without much consideration for the human element in the overall system.

However, investigations of the impact of new technology have shown that these assumptions are not tenable (they are what could be termed the substitution myth). Tasks and activities are highly interdependent or coupled in real complex systems.

Introduction of new automation has shifted the human role to one of monitor, exception handler, and manager of automated resources.

As a consequence, only some of these anticipated benefits of automation have, in fact, materialized -- primarily those related to the improved precision and economy of operations, i.e., those aspects of system operation that do not involve much interaction between human and machine. However, other expectations were not met, and unanticipated difficulties were observed. These problems are primarily associated with the fact that even highly automated systems still require operator involvement and therefore communication and coordination between human and machine. This need is not supported by most systems which are designed to be precise and powerful agents but are not equipped with communicative skills, with comprehensive access to the outside world, or with complete knowledge about the tasks in which it is engaged. Automated systems do not know when to initiate communication with the human about their intentions and activities or when to request additional information from the human. They do not always provide adequate feedback to the human who, in turn, has difficulties tracking automation status and behavior and realizing there is a need to intervene to avoid undesirable actions by the automation. The failure to design human-machine interaction to exhibit the basic competencies of human-human interaction is at the heart of problems with modern automated systems.

Another reason that observed difficulties with automation were not anticipated was the initial focus on quantitative aspects of the impact of modern technology. Expected benefits included reduced workload, reduced operational costs, increased precision, and fewer errors. Anticipated problems included the need for more training, less pilot proficiency, too much reliance on automation, or the presentation of too much information (for a more comprehensive list of automation-related questions see Wiener and Curry, 1980). Instead, it turned out that many of consequences of introducing modern automation technology were of a qualitative nature, as will be illustrated in later sections of this chapter. For example, task demands were not simply reduced but changed in nature. New cognitive demands were created, and the distribution of load over time changed. Some types of errors and failures declined whereas new error forms and paths to system breakdown were introduced.

Some expected benefits of automation did not materialize because they were postulated based on designers assumptions about intended rather than actual use of automation. The two can differ considerably if the future operating environment of a system is not sufficiently considered during the design process. In the case of cockpit automation, the actual and intended use of automation are not the same because, for example, air traffic control procedures do not match the abilities and limitations designed into modern flight deck systems and the various operators of highly advanced aircraft have different philosophies and preferences for how and when to use different automated resources.

Finally, design projects tend to experience severe resource pressure which almost invariably narrows the focus so that the automation is regarded as only an object or device that needs to possess certain features and perform certain functions under a narrowed range of conditions. The need to support interaction and coordination between the machine and its human user(s) in the interest of building a joint human-machine system becomes secondary. At this stage, potential benefits of a system may be lost, gaps begin to appear, oversimplifications arise, and boundaries are narrowed. The consequences are challenges to human performance.

Actual experiences with advanced automated systems confirm that automation does, in fact, have an effect on areas such as workload, error, or training. However, its impact turns out to be different and far more complex than anticipated. Workload and errors are not simply reduced, but changed. Modified procedures, data filtering or more-of-the-same training are not effective solutions to observed problems. Instead, the introduction of advanced automation seems to result in changes that are qualitative and context-dependent rather than quantitative and uniform in nature. In the following sections, some unexpected effects of automation will be discussed. They result from the introduction of automated systems that need to engage in, but were not designed for, cooperative activities with humans.

2 UNEXPECTED PROBLEMS WITH HUMAN-AUTOMATION INTERACTION

2.1 Workload - Unevenly Distributed, Not Reduced

The introduction of modern technology was expected to result in reduced workload. It turned out, however, that automation does not have a uniform effect on workload. As first discussed by Wiener (1989) in the context of modern technology for aviation applications, many automated systems support pilots most in traditionally low workload phases of flight but are of no use or even get in their way when help is needed most, namely in time-critical highly dynamic circumstances. One reason for this effect is the automation’s lack of comprehensive access to all flight-relevant data in the outside world. This leads to the requirement for pilots to provide automation with information about target parameters, to decide how automation should go about achieving these targets (e.g., selecting level and type of automated subsystem to invoke), to communicate appropriate instructions to the automation, and to monitor the automation closely to ensure that commands have been received and are carried out as intended. These task requirements do not create a problem during low workload phases of flight but once the descent and approach phases of flight are initiated, the situation changes drastically. Air traffic control (ATC) is likely to request frequent changes in the flight trajectory, and given that there is not (at this stage) a direct link between ATC controllers and automated systems, the pilot has the role of translator and mediator. He needs to communicate every new clearance to the machine, and he needs to (know how to) invoke system actions. It is during these traditionally high-workload, highly dynamic phases of flight that pilots report an additional increase in workload. Wiener (1989) coined the term "clumsy automation" to refer to this effect of automation on workload - a redistribution of workload over time rather than an overall decrease or increase because the automation creates new communication and coordination demands without supporting them well.

Workload is not only unevenly distributed over time but sometimes also between operators working as a team. For example, the pilot-not-flying on many advanced flight decks can be much busier than the pilot-flying as (s)he is responsible for most of the interaction with the automation interface which can turn a simple task (such as changing a route or an approach) into a “programming nightmare.”

The effect on workload was also unexpected in the sense that the quality rather than the quantity of workload is affected. For example, the operator’s task has shifted from active control to supervisory control by the introduction of automated systems. Humans are no longer continuously controlling a process themselves (although they still sometimes need to revert to manual control) but instead they monitor the performance of highly autonomous machine agents. This imposes new attentional demands, and it requires that the operator knows more about his systems in order to be able to understand, predict, and manipulate their behavior.

2.2 New Attentional and Knowledge Demands

The introduction of modern technology has created new knowledge and attentional requirements. Operators need to learn about the many different elements of highly complex systems and about the interaction of these elements. They need to understand input-output relationships to be able to anticipate effects of their own entries. In addition to knowing how the system works, they need to explore "how to work the system", i.e., operators must learn about available options, learn and remember how to deploy them across a variety of operational circumstances, and learn the interface manipulations required to invoke different modes and actions. Finally, it is not only the capabilities but also the limitations of systems that need to be considered.

Empirical research on human-automation interaction (e.g., Sarter and Woods, 1994a) has shown that operators sometimes have gaps and misconceptions in their model of a system. Sometimes operators possess adequate knowledge about a system in the sense of being able to recite facts, but they are unable to apply the knowledge successfully in an actual task context. This is called the problem of "inert" knowledge. One way to eliminate this problem is through training that conditionalizes knowledge to the contexts in which it is utilized.

Since the complexity of many modern systems cannot be fully covered in the amount of time and with the resources available in most training programs, operators learn only a subset of techniques or "recipes" to be able to make the system work under routine conditions. As a consequence, ongoing learning needs to take place during actual operations and has to be supported to help operators discover and correct bugs in their model of the automation. Recurrent training events can be used to elaborate their understanding of how the automation works in a risk-free environment.

Another problem related to knowledge requirements imposed by complex automation technology is that operators are sometimes miscalibrated with respect to their understanding of these systems. Experts are considered well calibrated if they are aware of the areas and circumstances for which they have correct knowledge and those in which their knowledge is limited or incomplete. In contrast, if experts are overconfident and wrongly believe that they understand all aspects of a system, then they are said to be miscalibrated (e.g., Wagenaar and Keren, 1986).

A case of operator miscalibration was revealed in a study on pilot-automation interaction where pilots were asked questions such as, "Are there modes and features of the Flight Management System (FMS) that you still don't understand?" (Sarter and Woods, 1994a; these kinds of questions were asked in an earlier study by Wiener, 1989). When their responses to this question are compared with behavioral data in a subsequent simulator study, there is some indication that these "glass cockpit" pilots were overconfident and miscalibrated about how well they understood the Flight Management System . The number and severity of pilots' problems during the simulated flight was higher than was to be expected from the survey. Similar results have been obtained in studies of physician interaction with computer based automated devices in the surgical operating room (Cook et al., 1991; Moll van Charante et al., 1993).

Several factors contribute to miscalibration. First, areas of incomplete or inaccurate knowledge can remain hidden from operators because they have the capability to work around these areas by limiting themselves to a few well practiced and well understood methods. In addition, situations that force operators into areas where their knowledge is limited and miscalibrated may arise infrequently. Empirical studies have indicated that ineffective feedback on the state and behavior of automated systems can be a factor that contributes to poor calibration (e.g., Wagenaar and Keren, 1986; Norman, 1990; Cook et al., 1991).

The need for adequate feedback design is related not only to the issue of knowledge calibration but also to the attentional demands imposed by the increased autonomous exhibited by modern systems. Operators need to know when to look where for information concerning (changes in) the status and behavior of the automation and of the system or process being managed or controlled by the automation. Knowledge and attentional demands are closely related, because the above mentioned mental model of the functional structure of the system provides the basis for internally guided attention allocation. In other words, knowing about inputs to the automation and about ways in which the automation processes these inputs permits the prediction of automation behavior which, in turn, allows the operator to anticipate the need for monitoring certain parameters. This form of attentional guidance is particularly important in the context of normal operations.

In case of anomalies or apparently inconsistent system behavior, it can be difficult or impossible for the user to form expectations. Therefore, under those circumstances, the system needs to provide external attentional guidance to the user to help detect and locate problems. The system interface needs to serve as an external memory for the operator by providing cues that help realize the need to monitor a particular piece of information or to activate certain aspects of knowledge about the system.

Two frequently observed ways in which attention allocation can fail is (a) a breakdown in the "mental bookkeeping" required to keep track of the multiple interleaved activities and events that arise in the operation of highly complex technology, and (b) a failure to revise a situation assessment in the presence of new conflicting information. In the latter case, called fixation error, evidence that is not in agreement with an operator's assessment of his situation is missed, dismissed, or rationalized as not really being discrepant.

The above problems -- gaps and misconceptions in an operator's mental model of a system as well as inadequate feedback design - can result in breakdowns in attention allocation which, in turn, can contribute to a loss of situation, or more specifically, system and mode awareness.

2.3 Breakdowns in Mode Awareness and "Automation Surprises"

Norman (1988, p. 179) explains device modes and mode error quite simply by suggesting that one way to increase the possibilities for error is to ". . . change the rules. Let something be done one way in one mode and another way in another mode." Mode errors occur when an intention is executed in a way appropriate for one mode when, in fact, the system is in a different mode. In simpler devices, each system activity was dependent upon operator input; as a consequence, the operator had to act for an error to occur.

With more advanced systems, each mode itself is an automated function that, once activated, is capable of carrying out long sequences of tasks autonomously in the absence of additional commands from human supervisors. This increased autonomy produces situations in which mode changes can occur based on situational and system factors. This capability for "indirect" mode changes, independent of direct and immediate instructions from the human supervisor, drives the demand for mode awareness. Mode awareness is the ability of a supervisor to track and to anticipate the behavior of automated systems (Sarter and Woods, 1995).

Breakdowns in mode awareness result in "automation surprises." These automation surprises have been observed and reported in various domains (most notably flightdeck and operating room automation, e.g., Sarter and Woods, 1994a; Moll van Charante et al., 1993) and have contributed to a considerable number of incidents and accidents. Breakdowns in mode awareness can led to mode errors of omission in which the operator fails to observe and intervene with uncommanded and/or undesirable system behavior.

Early automated systems tended to involve only a small number of modes that were independent of each other. These modes represented the background on which the operator would act by entering target data and by requesting system functions. Most functions were associated with only one overall mode setting. Consequently, mode annunciations (indications of the currently active as well as planned modes and of transitions between mode configurations) were few and simple and could be shown in one central location. The consequences of a breakdown in an operator's awareness of the system configuration tended to be small, in part because of the short time-constant feedback loops involved in these systems. Operators were able to detect and recover from erroneous input relatively quickly.

The flexibility of more advanced technology allows and tempts automation designers to develop much more complex mode-rich systems. Modes proliferate as designers provide multiple levels of automation and various methods for accomplishing individual functions. The result is a large number of indications of the status and behavior of the automated system(s), distributed over several displays in different locations. Not only the number of modes but also, and even more importantly, the complexity of their interactions has increased dramatically.

The increased autonomy of modern automated systems leads to an increase in the delay between user input and feedback about system behavior. These longer time-constant feedback loops make it more difficult to detect and recover from errors and challenge the human's ability to maintain awareness of the active and armed modes, the contingent interactions between environmental status and mode behavior, and the contingent interactions across modes.

Another contributing factor to problems with mode awareness relates to the number and nature of sources of input that can evoke changes in system status and behavior. Early systems would change their mode status and behavior only in response to operator input. More advanced technology, on the other hand, may change modes based on sensor information concerning environment and system variables as well from input by one or multiple human operators. Mode transitions can now occur in the absence of any immediately preceding user input. In the case of highly automated cockpits, for example, a mode transition can occur when a preprogrammed intermediate target (e.g., a target altitude) is reached or when the system changes its mode to prevent the pilot from putting the aircraft into an unsafe configuration.

Indirect mode transitions can arise in another way as side effects of a direct operator input to automated systems. This potential is created by the fact that the effects of operator input depend on the status of the system and of the environment at the time of input. The user intends one effect but the complexity of the interconnections between different automated subsystems and modes may mean that other un-intended changes occur automatically. Thus, an action intended to have one particular effect can have a different effect or additional unintended side effects due to automation designs that increase system coupling. missing these side effects is a predictable error form that is accentuated because of weak feedback concerning mode status and transitions and when there are gaps or misconceptions in user's mental model of the system. Incidents and accidents have shown that missing these side effects can be disastrous in some circumstances.

2.4 New Coordination Demands

When new automation is introduced into a system or when there is an increase in the autonomy of automated systems, developers often assume that adding "automation" is a simple substitution of a machine activity for human activity (the substitution myth). Empirical data on the relationship of people and technology suggests that is not the case. Instead, adding or expanding the machine's role changes the cooperative architecture, changing the human's role often in profound ways. Creating partially autonomous machine agents is, in part, like adding a new team member. One result is the introduction of new coordination demands. When it is hard to direct the machine agents and hard to see their activities and intentions, it is difficult for human supervisors to coordinate activities. This is one factor that may explain why people "escape" from clumsy automation as task demands escalate. Designing for coordination is a post-condition of more capable machine agents. However, because of the substitution myth, development projects rarely include specific consideration how to make the automation an effective team player or evaluation of possible systems along this dimension.

One concrete example of coordination occurs when automation compensates for a fault but only up to a point. When the automation can no longer handle the situation, it gives up and turns the problem back over to the human crew. The problem is that this transfer control can easily be far from bumpless. The people may not be aware of the problem or may not fully understand the developments up to that point if the automation compensates for the fault silently. Suddenly, when the effects of the fault are already substantial, they are forced to take over control. This is a challenging situation for them that has contributed to several accident sequences in aviation and to incidents in anesthesiology.

The above case is an example of a decompensation incident (Woods, 1994). Decompensation incidents in managing highly automated processes are one kind of complication that can arise when automatic systems respond to compensate for abnormal influences generated by a fault. As the abnormal influences produced by the fault persist or grow over time, the capacity of the automation's counter-influences to compensate becomes exhausted. When the automation's capacity to counteract is exhausted, it hands control back to human team members. However, they may not be prepared to deal with the situation (they may not appreciate the seriousness of the situation or may misunderstand the trouble) or the situation may have progressed too far for them to contribute constructively.

The presence of automatic counter-influences leads to a two phase signature. In phase 1 there is a gradual falling off from desired states over a period of time. Eventually, if the practitioner does not intervene in appropriate and timely ways, phase 2 occurs -- a relatively rapid collapse when the capacities of the automatic systems are exceeded or exhausted. During the first phase of a decompensation incident, symptoms may not be present or may be small, which makes it difficult for human supervisors to understand what is occurring. This can lead to great surprise when the second phase occurs

In those situations, the critical information for the human operator is not the symptoms per se but the force with which they must be resisted. An effective human team member would notice and communicate the need to exert unusual control effort (Norman, 1990). Thus, lack of information about automatic system activities can contribute to the failure to recognize the seriousness of the situation and the failure of the supervisory controller to act to invoke stronger counter-actions early enough to avoid the decompensation.

This example also can show us what it means to provide effective feedback. But what kind of feedback will prove effective? We could address each specific example of the need for better feedback about automation activities individually, one at a time. But this piecemeal approach will generate more displays, more symbolic codings on displays, more sounds, more alarms. More data will be available, but these will not be effective as feedback because they challenge the crew's ability to focus on and digest what is relevant in a particular situation. Instead, we need to look at the set of problems that all point to the need for improved feedback to devise an integrated solution. For example, the decompensation signature show us a class of feedback problems that arise when automation is working at the extreme of its envelope or authority. The automation doesn't clearly tell the human supervisor that this is the case; when control passes back to people, they are behind the developing situation and the transfer is rather bumpy or worse.

For this class of cases new feedback and communication between the human and machine agents is needed to indicate:
- when I (the automation) am having <u>trouble</u> handling the situation;
- when I (the automation) am taking <u>extreme</u> action or <u>moving towards</u> the extreme part of my authority.

This specifies a performance target. The design question is how to make the system smart enough to communicate this intelligently? How to define what are "extreme" regions of authority in a context sensitive way? When is an agent having trouble in performing a function (but not yet failing to perform)? How does one effectively communicate moving toward a limit rather than just invoking a threshold crossing alarm?

From experience and research we know some constraints on the answers to these questions. Threshold crossing indications (simple alarms) are not smart enough -- thresholds are either set too late or too early. We need a more gradual escalation or staged shift in level or kind of feedback. We know that providing an indication whenever there is any automation activity (e.g., auditory signal) says too much, too soon. We want to indicate <u>trouble</u> in performing the function or <u>extreme</u> action to accomplish the function, not simply any action.

We know that there are certain errors that can occur in designing feedback that should be avoided in this case. We must avoid

- nuisance communication such as voice alerts that talk to you too much in the wrong situations,
- excessive false alarms,
- distracting indications when more serious tasks are being handled (e.g., a warning on constantly or at a high noise level during a difficult situation -- "silence that thing!").

In other words, misdesigned feedback can talk too much, too soon or it can be too silent, speaking up too little, too late as automation moves toward authority limits.

Should the feedback occur visually or through the auditory channel or through multiple indications? Should this be a separate new indication or integrated into existing displays? Should the indication be of very high perceptual salience? In other words, how strongly should the signal capture user attention? Working out these design decisions requires developing prototypes and adjusting the indications in terms of perceptual salience, along a temporal dimension (when to communicate), and along a strength dimension (how gabby) based on human performance in context. All of which requires thinking about these new indications in the context of other possible signals.

2.5 The Need for New Approaches to Training

With the introduction of highly advanced automation technology, traditional approaches to training no longer seem adequate to prepare operators for their new task of supervisory control of highly dynamic and complex systems. The aviation domain is one area where this problem became highly noticeable in the 1980s. Failure rates in transition training for "glass cockpit" aircraft were at an all time high (Wiener, 1993), and even today the introduction of a new highly advanced airplane to an airline's fleet tends to create challenges and problems. Whereas today most pilots successfully complete their training, they still report that the first transition to one of these new airplanes is considerably more difficult and demanding than the transition between two conventional aircraft.

The observed problems are interpreted by many as indications of a need for more training time to acquire more knowledge about these complex systems. However, training-oriented research in other similarly complex domains (e.g., Feltovich et al., 1991) and a better understanding of the kinds of problems experienced by 'glass cockpit' pilots suggest that it is the nature of training that needs to be reconsidered rather than its duration. Cockpit technology has changed in fundamental ways that require new ways of learning and practice. It is no longer possible to learn about these systems by accumulating compartmentalized knowledge about individual components and simple input-output relations. Instead, pilots need to form a mental model of the overall functional structure of the system to understand its contingencies and interactions. Such a model is a prerequisite for being able to monitor and coordinate activities with cockpit automation as was explained in the earlier section on attentional demands imposed by modern automated systems.

A mental model helps build expectations of system behavior, and it contributes to the adequate allocation of attention across and within numerous information-rich cockpit displays. It also supports pilots in dealing with novel situations by allowing them to derive possible actions and solutions based on their general understanding of how the system works (Carroll and Olson, 1988). These affordances of a mental model make it the desirable objective of training for advanced cockpit systems, which no longer allow for exposure to all aspects of their operation during training -- even with more training time.

To support the formation of an effective mental model, training needs to encourage pilots to explore actively the available options and dynamics of the automation. Inventing a model based on experimentation has been shown to be preferable to the explicit teaching of a system model (Carroll and Olson, 1988). As Spiro et al. (1988, p. 378) point out, "knowledge that will be used in many ways has to be learned, represented, and tried out in many ways." In contrast, rote memorization is antithetical to the development of applicable knowledge, i.e., knowledge that can be activated in context. Learning recipes results in "inert" knowledge where the user can recite facts but fails to apply this knowledge effectively in actual line operations (see Feltovich et al., 1991).

In summary, part of the solution to observed problems with flying highly automated aircraft may be new approaches to and objectives for training rather than simply more training time. But it is important to realize that even such improved training cannot solve all observed problems -- training cannot and should not be a fix for bad design.

2.6 New Opportunities for New Kinds of Error

Another anticipated benefit of automation was a reduction in human error -- but again, operational experience and systematic empirical research proved otherwise. Instead of reducing the overall amount of errors, automation provided new opportunities for different kinds of error (Woods et al., 1994). One example that has recently gained considerable interest is the case of mode awareness and its relationship to automation surprises, especially in the context of new flightdecks.

Interest in research on mode error on glass cockpit aircraft was triggered by the results of a study by Wiener (1989) who conducted a survey of B-757 pilots where about 55% of all respondents said that they were still being surprised by the automation after more than one year of line experience on the aircraft. In a follow-up study, Sarter and Woods (1992) sampled a different group of pilots at a different airline who were flying a different glass cockpit aircraft (the B737-300/400). They replicated Wiener's results and, more importantly, they gathered detailed information concerning the nature of and underlying reasons for ‘automation surprises’, which can be seen as symptoms of a loss of mode awareness, i.e., awareness of the status and behavior of the automation. In addition, an experimental simulator study was carried out to assess pilots’ mode awareness in a more systematic way (Sarter and Woods, 1994a). Overall, this research confirmed that "automation surprises" are experienced even by pilots with a considerable amount of line experience on highly automated aircraft. Problems with mode awareness were shown to occur most frequently in non-normal and time critical situations.

Mode errors seem to occur because of a combination of gaps and misconceptions in operators’ model of the automated systems and the failure of the automation interface to provide users with salient indications of its status and behavior (Sarter and Woods, 1994a, 1995). During normal operations, bugs in operators’ mental model can make it difficult or impossible to form accurate expectations of system behavior that provide guidance for effective allocation of attention across and within displays. In the case of non-normal events that cannot be anticipated by the crew, the automation interface needs to attract the user's attention to the relevant piece(s) of information -- many current systems fail to do so.

Mode errors of omission are particularly disturbing because of the implications for error recovery. Mode errors of omission, however, occur in the absence of an immediately preceding directly related pilot action or input. Therefore, operators are less likely to notice a change in system status or behavior. The trend toward errors of omission also has implications for the development of countermeasures to mode error. One proposal for dealing with mode errors of commission has been to introduce forcing functions that prevent the user from carrying out certain actions or inputs. In the case of errors of omission, however, such measures would not be useful as there is no specific action to prevent.

It seems that the detection of unexpected and possibly undesired changes in automation behavior is one of the major difficulties with the coordination between humans and advanced automated systems. The problem is difficult because the task is not to detect a state or behavior that is always abnormal. Instead the task is to detect that a certain system behavior, which may be normal and acceptable in some circumstances, requires operator intervention in this context.

2.7 Complacency and Trust in Automation

Complacency has been proposed as another factor contributing to operators' failure to detect and intervene with system failures or undesirable system behavior. Complacency refers to the development of a false sense of security as operators come to rely on automation which is, in fact, highly reliable but can still fail without warning (Billings, 1991; 1996).

This view has raised concerns as it seems to blame the human by implying a lack of motivation and concentration on the task at hand. It seems to suggest that if the human would only try harder, it would be possible for him to perform his duties successfully. It fails to acknowledge that people will come to rely on systems that appear to be reliable at least for high frequency of encounter situations. The design of the joint human-machine system has created a role where people must monitoring for rare events -- a sustained attention task. Complacency may be more indicative of the need to rethink the human-machine architecture.

Trust in automation is a related issue that has received considerable consideration in the literature on human-automation interaction (e.g., Muir, 1987). Trust miscalibration is a problem associated with current increasingly autonomous and powerful systems that can create the image of a highly intelligent and proficient partner. What do we mean by "trust" in the context of human-machine interaction? One recent definition of trust (Barber, 1983) includes two important dimensions, namely the expectation of technically competent role performance and the expectation that a partner will carry out his fiduciary obligations and responsibilities, i.e., his duty to place, in certain situations, other's interests before his own. This latter form of trust is important in situations where it is not possible for the user to evaluate the technical competence of another agent. In those cases, the user has to rely on the moral obligation of the other agent not to misuse the power given to him/it. Muir (1987, p. 530) has predicted that "the issue of machine responsibility will become more important in human-machine relationships to the extent that we choose to delegate autonomy and authority to 'intelligent', but prosthetic machines. The more power they are given, the greater will be the need for them to effectively communicate the intent of their actions, so that people who use them can have an appropriate expectation of their responsibility and interact with them efficiently."

Finally, trust in automation is related to the issue of responsibility. Jordan was one of the first to point out that "we can never assign them [i.e., the machines] any responsibility for getting the task done; responsibility can be assigned to man only" (Jordan, 1963, p. 164). Therefore, he required that responsibilities be clearly assigned to each human in the system and that each human be provided with the means necessary to effectively control the tasks and systems for which he is responsible. Jordan's point that responsibility cannot be assigned to a machine has been expanded on by Winograd and Flores (1986) who point out that one of the major differences between human-human interaction and human-machine interaction is that we may treat an intelligent machine as a rational being but not as a responsible being. "An essential part of being human is the ability to enter into commitments and to be responsible for the courses of action that they anticipate. A computer can never enter into a commitment (although it can be a medium in which the commitments of the designers are conveyed), ..." (Winograd and Flores, 1986, p. 106). The last sentence of this statement is important because it suggests that designers may have to be held responsible for the machines they build. Just because machines can not be responsible may not mean that their operators have to bear all responsibility alone. Designers need to ensure that their systems comply with requirements of human-centered automation such as being predictable, accountable, and dependable (see Billings, 1996), all of which are aspects of responsibility.

In the preceding paragraphs, unanticipated problems associated with the design of and training for modern automated systems have been discussed. It was shown that anticipated benefits and disadvantages of automation were often conceived of in quantitative terms - less workload, more precision, less errors, more training requirements - but that observed problems tended to be qualitative in nature - new temporal workload patterns, different kinds of errors and failure patterns, the need for different approaches to training. Observed problems with human-automation interaction are related to the need for, but lack of support for, communication and coordination between human operators and machine agents. This, in turn, is the consequence of increasingly high levels of system complexity, coupling, autonomy, and authority in combination with low system observability (Sarter and Woods, 1994b; Woods, 1996). In the following section, a number of accidents involving modern technology are presented that illustrate how these system properties can lead to breakdowns in overall system performance.

3 ACCIDENTS INVOLVING BREAKDOWNS IN HUMAN-MACHINE COORDINATION

In the following section, three accidents involving breakdowns in the communication and coordination between human and machine agents in various domains are described. They illustrate how a combination of several of the above discussed factors, as well as additional factors such as organizational pressures, can contribute to the evolution of disastrous events.

3.1 Radiation Accidents with Therac-25

Another series of accidents related to human-machine interaction occurred in the medical world with a system called Therac-25, a computerized radiation therapy machine for cancer therapy. The machine can be used to deliver a high-energy electron beam to destroy tumors in relatively shallow tissue areas; deeper tissue can be reached by converting the electron beam into x-ray photons.

In the mid 1980s, several accidents happened with this device, all of which involved a lack of feedback about the status and activities of the machine. One of those accidents, which has been analyzed in quite some detail, occurred in 1986 at the East Texas Cancer Center. In this case, a patient was undergoing his ninth treatment as a follow-up to the removal of a tumor from his back. During this procedure, the technician operating the device would normally be in contact with the patient via a video camera and the intercom in the treatment room. On this particular day, however, both these sources of feedback were temporarily inoperative. Another thing happened that day that had not happened before. The technician made a mistake when setting up the device for the treatment. She entered an "x" (the entry for x-ray treatment) instead of an "e" (for electron mode), realized her mistake after a few more entries, and tried to correct it quickly by selecting the "edit" function of the display and by entering an "e" in place of the "x". Then she hit the return key several times to leave all other entries unchanged.

What the technician did not know and could not see was that the machine did not accept her corrections despite the fact that the screen displayed the correct therapy entries. The particular sequence of keystrokes she entered plus the speed she entered them (less than eight seconds because she was proficient with the interface) had revealed a software design problem. When the machine was activated by the technician, the machine paused and a display indicated to her "Malfunction 54" and "treatment pause." This general feedback was familiar to her; it indicated that the treatment had not been initiated because of some minor glitch. On a separate documentation sheet, she found the error number explained as a "dose input 2" error but the technician did not understand the meaning of this message. In general the machine paused when a minor glitch occurred; when there were more significant problems, in her experience the machine abandoned the treatment plan and reverted to an initial state. Being used to many quirks of the machine, she decided simply to activate the beam again. When she re-activated the device, the same sequence of events occurred.

Meanwhile, inside the treatment room, unbeknown to the technician, the patient was hit by a massive radiation overdose first in the back and then on his arm as he tried to get up from the treatment table after the first painful episode.

After the patient reported intense pain during and after the treatment, the equipment was examined but no malfunction was diagnosed and nothing physically wrong was found with the patient. As no problems could be identified and because other patients were waiting in line, they started using the equipment again the same day. The patient died a few months later from complications related to the overdose he received on this day (for a detailed description and analysis of this accident see Leveson and Turner, 1993).

This case illustrates how a lack of feedback concerning both the affected process (the display of dosage could not go high enough to indicate the dosage received) and the status and behavior of the automated system (the gap between what the machine said it would do and what instructions it was actually following; the pause type alarm was a familiar occurrence and minor issue) make it impossible for the operator to detect and intervene with undesirable system behavior. It also shows how organizational pressures such as the large number of patients waiting to be treated with the Therac-25 device added to the problem faced by the operator and prevented a timely investigation of the device. And finally that virtually all complex software can be made to behave in an unexpected fashion under some conditions. (Leveson and Turner, 1993).

3.2 A Fatal Test Flight

This accident occurred in the context of a test flight of one of the most advanced automated aircraft in operation. It involved a simulated engine failure at low altitude under extreme flight conditions. A number of things were out of the ordinary on this flight, and would later be presented as contributing factors in the accident investigation report (for a detailed account of this accident see Aviation Week and Space Technology, April 3, April 10, and April 17, 1995). During takeoff, the co-pilot rotated the aircraft rather rapidly which resulted in a pitch angle of slightly more than 25 degrees within 6 seconds after takeoff. At that point, the autopilot was engaged as planned for this test. Immediately following the autopilot engagement, the Captain brought the left engine to idle power and cut off one hydraulic system to simulate an engine failure situation. This particular combination of actions and circumstances led up to a crash killing everyone aboard the aircraft which was totally destroyed. How did this happen?

When the autopilot was selected, it immediately engaged in an altitude capture mode because of the high rate of climb and the rather low pilot-selected level-off altitude of 2,000 ft. At the same time, because the pitch angle exceeded 25 degrees at that point, the declutter mode of the Primary Flight Display activated. This means that all indications of the active mode configuration of the automation (including the indication of the altitude capture mode) were hidden from the crew because they had been removed from the display for simplification.

Another important factor in this accident is the inconsistency of the automation design in terms of protection functions, which are intended to prevent or recover from unsafe flight attitudes and configurations. One of these protection functions guards against excessive pitch which results in too low an airspeed. This protection is provided in all automation configurations except one - the very altitude acquisition mode in which the autopilot was operating. As a consequence, the automation continued to try to follow an altitude acquisition path even when it became impossible to achieve it (after the Captain had brought the left engine to idle power). Ultimately, the automation flew the aircraft into a stall, and the crew was not able to recover given the low altitude.

Clearly, a combination of factors contributed to this accident and was cited in the report of the accident investigation. Included in the list of factors are the extreme conditions under which the test was planned to be executed, the lack of pitch protection in the altitude acquisition mode and the inability of the crew to determine that the automation had entered that particular mode because of the decluttering of the Primary Flight Display. The time available for the Captain to react (12 seconds) to the abnormal situation was also cited as a factor in this accident.

A more generic contributing factor in this accident was the behavior of the automation which was highly complex, inconsistent and difficult to understand. These characteristics made it hard for the crew to anticipate the outcome of the maneuver. In addition, the observability of the system was practically non-existent when the declutter mode of the Primary Flight Display activated upon reaching a pitch angle of more than 25 degrees up.

These problems and accidents are clearly related to and call for changes in the design of highly automated systems in the interest of making them more observable and cooperative agents. To achieve this goal, various approaches to system design have been proposed and are discussed in the following section.

4 DIFFERENT APPROACHES TO AUTOMATION DESIGN

The problems and accidents with automated systems described in the previous sections are to a large extent the result of technology-centered design that does not consider the need for supporting communication and cooperation between human and machine agents. To counteract and prevent the re-occurrence of such difficulties, a new approach, called human-centered automation, has been developed. The following sections provide an overview of the basic principles underlying this new perspective. We will also discuss extensions of the basic human-centered view to consider supporting not just one but a group of human users and their interaction and to consider integrating humans, machines, and their task environments in an effort to improve overall system performance.

4.1 Human- vs. Technology-Centered Automation

The unanticipated problems associated with clumsy automation has led to the call for human-centered rather than technology-centered automation. What do we mean by these labels? Norman illustrated the difference by quoting and re-writing the motto of the Chicago World's Fair (1933). The original was, "Science finds, industry applies, man conforms." In contrast, Norman (1994) suggested, "people propose, science studies, technology conforms" as a human-centered alternative.

In a technology-centered approach the primary focus is technological feasibility -- what is needed to create machines that can function more autonomously. Human-centered automation, on the other hand, is oriented toward operational needs and practitioner requirements. Its objective is to support, not supplant or replace the human operator. The primary focus becomes how to make automated system team players.

Basically, in a user-centered approach designers consider, up front, the impact of introducing new technology and automation on the role of people in the system and on the structure of the larger system of which the automation is a part. This approach is needed because of technological success: the dominant question is rarely what can be automated but what should be automated in support of human operators and of human-machine cooperation.

It is very important to be clear that human-centered automation is <u>not</u> a call for less technology. In contrast, it calls for developing high technology that is adapted to the pressures of the operational world. People have always developed and skillfully wielded technology as a tool to transform and amplify their work, whether physical work or cognitive work. As Norman (1988) puts it, "technology can make us <u>smart</u> and technology can make us <u>dumb</u>" (emphasis added). The central problem is not less or more technology, but rather skillful or clumsy technology.

Guiding principles for a human-centered approach to design have been put forward by Billings (1991), who suggests that as long as human operators bear ultimate responsibility for operational goals, they must be in command. To be in command effectively, operators need to be involved in and informed about ongoing activities and system states and behaviors. In other words, the automation must be observable, and it needs to act in predictable ways (see Billings 1991, 1996 for a comprehensive discussion of human-centered automation).

Let us illustrate the difference between human- and technology-centered perspectives by examining some recent incidents and accidents in the aviation domain. Some of these events involved pilots who were trying to take control of an airplane upon observing unexpected or undesirable automation behavior. They attempted but failed to disengage or overpower the automation, thus creating a "fight" between man and machine over the control of the aircraft -- in some cases with fatal consequences (for examples see Dornheim, 1995).

The technology-centered view of these events puts people and machines into opposition. Technologists assert that the problem was caused by the inappropriate behavior of a pilot who interfered with the activities of an automated system that acted as designed and as instructed earlier by the same pilot. The only alternative, from a technology-centered point of view is to say that the machine was to blame. The problem must be either in the people or in the machine.

In contrast, a human-centered approach shifts the boundaries. Human and machine agents are together part of one system. The kinds of accidents mentioned earlier reveal a breakdown in coordination between the human and machine portions of a team. Solutions involve developing the mechanisms to produce improved team play between machine and human agents just as we have recognized that effective team play is critical for the success of crews of human operators.

The two perspectives also try to address the problem in very different ways. The technology-centered approach suggests efforts to modify people -- remedial training, new procedures, more technology intended to eliminate human activity. In contrast, the human-centered view focuses on changing the human-machine system in ways to support better coordination. Should a automated subordinate continues to act in opposition to the pilot's latest input or is this an act of insubordination? Should an effective subordinate communicate with their supervisor pointing out conflicting inputs and asking for clarification?

Being in command is possible only if one is provided with or has access to all information necessary to assess the status and behavior of the automation and to make decisions about future courses of action based on this assessment. A major objective of keeping the operator informed is to avoid undermining his or her authority (for a broader discussion of these issues see Billings, 1996). If information is hidden from the operator or not provided except in a hazardous situation, he is forced into a reactive mode.

Authority also implies that the operator has means to instruct, re-direct and if need be 'escape' from the automation when deemed necessary. Having available the nominal means to instruct or escape is not sufficient. These mechanisms have to be usable under actual task conditions of multiple tasks, a dynamic world, and multiple data sources competing for attention. If, for example, control over the automation can be achieved only by means of a sequence of rarely executed actions that may require the diversion of attention away from critical system parameters in an escalating problematic situation, the automation is only a burden and not a resource or support.

4.2 Automation Design--User-Centered, Crew-Centered, Practice-Centered?

The original label for the perspective outlined in the previous section and introduced by several authors was human-centered (or user-centered). This label has some limits. For example, it seems to suggest that there is an individual human who should be supported by new designs. It turns out, of course, that rarely is the issue a single individual. Perhaps the label should be team-centered. Some may think this individual human is whoever does the task today. Technology change does transform the roles people play; a human-centered process focuses on supporting the new roles of people as supervisory controllers, exception handlers and monitors and managers of automated resources. Others may object that our goal is to improve system performance and not merely to "enrich" an individual's or team's single job. Emphasizing the integration across many practitioners, instruments, and tasks leads some to prefer such labels as "use-centered" or "practice-centered" (e.g., Flach and Dominguez, 1995). Use-centered means that (a) we are trying to make new technology sensitive to the constraints and pressures acting in the actual operational world, and (b) we are focused on multiple actors at different levels with different scopes of responsibility embedded in a larger operational system. Some of these agents are human and some machines. We need to think of new automation as part of this control and management system rather than simply divide the world into machine and human parts. We need to design this system of interacting control and management agents to perform effectively given the demands of the domain.

Human-, team-, practice-centered, all of the labels center design on someone or something. But all point to an overall systems perspective that implies that no single element of the system should be at the center of designers' considerations. Instead, all system components are viewed as mutually dependent and interacting with one another. All of them involve constraints as well as the potential for change within certain limits. The integration of those constraints is the design objective.

4.3 The Gap Between User-Centered Intentions and Actual Practice

Interestingly, the need for a "human-centered" approach to automation design is accepted by many people involved in the design and evaluation of modern technology. For example, almost all parties in the aviation industry agree that human-centered automation is an appropriate goal. Yet, as we have discussed, flight deck automation is a well studied case in which many automation designs exhibit problems in human-automation coordination. Hence the epigraph that opens this chapter. This gap between human-centered intentions and human-centered development practices indicates there is either a misunderstanding about the concept of human-centeredness or an inability to translate its underlying ideas into actual designs.

Gaps between user centered intentions and actual design practices arise in part because the developer of new technology and the people who must use it in actual work have two different perspectives. The developers' eye view is apparent simplicity. They justify their new technology in terms of potential benefits, often benefits derived from their claims about how the new technology will affect human performance. The practitioners' eye view is real complexity. They see all of the complicating factors that can arise in the operational world at the margins of normality and in more exceptional circumstances. They experience the new burdens created by clumsy use of technology. They must, as responsible agents, make up for the gaps between the developer's dreams and the actual complexities of practice.

The gap between user-centered intentions and technology-centered development occurs when designers:

- oversimplify the pressures and task demands from the users' perspective,
- assume that people can and will call to mind all relevant knowledge,
- are overconfident that they have taken into account all meaningful circumstances and scenarios,
- assume that machines never err,
- make assumptions about how technology impacts on human performance without checking for empirical support or despite contrary evidence,
- define design decisions in terms of what it takes to get the technology to work,
- sacrifice user oriented aspects first when tradeoffs arise,
- focus on building the system first, then trying to integrate the results with users.

5 AUTOMATION - A WIDE RANGE OF TOOLS AND AGENTS

Taking a human-centered and system-oriented approach to the design and evaluation of modern technology requires that one aim at identifying and integrating the constraints associated with the human user(s), the automation, and the task (environment). These constraints are changing over time. Therefore, mismatches may disappear and new ones may be created.

One important system element that is undergoing considerable changes is the machine element. Increasingly sophisticated automated systems are being introduced to a wide range of domains where they can serve a variety of purposes. Automated systems can support or take over control of subtasks, they can process and present information, or they can carry out system management tasks (Billings, 1996).

Because of the different demands associated with these tasks and functions, and also due to the continuing evolution of computational power and of automation philosophies, automated systems differ to a considerable extent. It may therefore not be appropriate to use the term "automation" as though it referred to one class of homogenous systems. Instead, these systems differ with respect to many important properties that affect the relationship between the system and its human user(s).

Examples of such properties are a system's level of authority and autonomy as well as its complexity, coupling, and observability (Woods, 1996). The term "authority" refers to the power to control a process. The level of authority of an automated control system has implications for the role and responsibility assigned to its human operator. "Autonomy" denotes a system's capability to carry out sequences of actions without requiring (immediately preceding) operator input. In other words, autonomy refers to a system's level of independence from the human user for some specific task. System complexity is determined by the number of system components and especially by the extent and nature of their interactions. Coupling refers to the potential for an event, fault, or action to have multiple cascading effects. The higher the level of autonomy, complexity, and/or coupling of a system, the greater is the need for communication and coordination between human and machine to support the operator's awareness of the state and behavior of automation.

System awareness is the prerequisite for realizing the need for intervention with system activities that are not desirable or may even be dangerous. The key to supporting human-machine communication and system awareness is a high level of system observability. Observability is the technical term that refers to the cognitive work needed to extract meaning from available data (Rasmussen, 1985). This term captures the fundamental relationship among data, observer and context of observation that is fundamental to effective feedback. Observability is distinct from data availability, which refers to the mere presence of data in some form in some location. Observability refers to processes involved in extracting useful information. It results from the interplay between a human user knowing when to look for what information at what point in time and a system that structures data to support attentional guidance.

To summarize, automated systems are not homogeneous but rather differ and continue to change along a number of important dimensions that can affect the interaction between humans and machines. One prerequisite for making progress in understanding and supporting the coordination between these two agents is to define them both in terms of their abilities, strategies, and limitations. Just as we consider differences between human operators such as the different abilities and strategies of novices versus experts, we need to better define the nature of automated systems because it is the degree to which human and machine properties match and support each other that determines to a large extent the overall system performance.

6 AUTOMATION DESIGN - CURRENT TRENDS AND FUTURE NEEDS

6.1 Ongoing Trends in Automation Design

Automated systems differ significantly as the result of a continuous evolution of technological capabilities in combination with the different automation philosophies that determine how these capabilities are utilized and implemented.

One trend in automation design is toward higher levels of system autonomy, authority, complexity, and coupling. These properties create an increased need for communication and coordination between humans and machines. To support this need, the design of system feedback would have to be improved in the interest of providing the automation with communicative skills. This need has not been sufficiently addressed, however. The amount of information that is potentially available to the operator has increased; but its quality does not match the mechanisms and limitations of human information processing. As a consequence, the gap between available and required feedback is growing. This has been shown to have a negative impact on the ability of operators to maintain awareness of automation status and behavior and to coordinate their activities with these advanced systems (e.g., Sarter and Woods, 1995a; 1995b).

6.2 The Need for Cooperative Systems

When unexpected problems followed new automated systems, they have been explained in two different ways. One group of commentators blamed problems on the human element in the system. They stated that the machines worked as intended, and that it was the variability in the operator’s performance or "human error” that caused the problem. This argument was and is still being made by those who think that the solution to automation-related difficulties is even more automation. The very same problems have been explained by others as evidence of "over-automation."

Research that has examined the effects of new automation on human performance indicates that both of these reactions are too simple (e.g., Norman, 1990). The data indicates that the problems are associate with breakdowns in coordination of human and machine agents. These coordination breakdowns follow this general template. An event occurs or set of circumstances come together that appears to be minor, at least in principle. This initial event or action triggers an evolving situation from which it is possible to recover. But through a series of commissions and omissions, misassessments and miscommunications, the human and automation team manage the situation into a much more serious and risky incident or even accident. The mismanagement hinges on the misassessments and miscommunications between the human and machine agents. It is results of this kind that have led to the recognition that machine agents cannot be designed merely as strong individual agents but need to be designed so as to support coordinated activity across the team. Similarly, the human supervisory role requires new knowledge and skills to manage automated resources effectively across a variety of potential circumstances.

Thus, the critical unit of analysis is the joint human-machine system. Success depends on supporting effective communication and coordination between humans and increasingly autonomous and powerful systems that can act independently of operator commands. This emphasizes the need to identify and integrate the constraints associated with machines, humans, tasks, and the environment in which they cooperate in order to increase the efficiency and safety of operations.

As other domains begin to introduce new levels of automation, they can avoid the problems that have occurred in the past by considering how to make automated systems team players early in the development process. One domain where the new levels of automation are likely to be introduced in the near future is air traffic control. Plans for future air traffic management are based on the idea to provide airspace users with more flexibility and authority in order to increase the efficiency of air carrier operations and the capacity of the air traffic system. This increased flexibility will reduce the potential for long-term planning which forms the basis for current controller decisions and interventions. New airborne and ground-based automated systems will have to be introduced to support a more short-term approach to the safe handling of traffic.

The first challenge created by the introduction of more and of more complex and powerful systems will be to ensure that these systems not only perform their assigned tasks of traffic separation and guidance but that they communicate to their users about their status, reasoning, and behavior. For some of the systems that will still exist in the future environment, it has already been shown that breakdowns in the interaction between humans and these machines occur because it can become very difficult for the user to keep track of his "strong but silent" counterpart. It is not known how the addition of more systems, the large number of human and machine agents, and the potential for machine-machine communication may affect the overall safety and efficiency of this system.

6.3 Networks of Automated Systems

Increasingly, networks of automated systems are envisioned and created to handle highly complex tasks and situations by engaging in negotiations and coordination among themselves without a need for direct involvement of the human operator. Such machine networks may become part of future air traffic management where, for example, ground-based computers may talk to airborne Flight Management Computers to negotiate and communicate changes in flight plans.

The challenge will be to develop technology that contributes to the communication and coordination among all human and machine players involved in this distributed network of decision-makers. It will be critical to avoid additional management, monitoring, and coordination tasks for the human operator resulting from "clumsy" and "silent" implementation of automation.

7 CONCLUSION

The introduction of advanced technology has created automation surprises: system operators are surprised by the behavior of their strong but silent machine partners; system designers/purchasers are surprised to find new problems that concern the coordination of people and automated systems. Practitioners have to cope with resulting breakdowns in user-system coordination and with uncommanded, unanticipated, and sometimes undesirable activities of their machine counterparts. Designers are faced with unexpected consequences of the failure to support communication adequately between humans and machines. These surprises are not simply the result of over-automation or human error. Instead they represent a failure to design for a coordinated team effort across human and machine agents as one cooperative system.

This design failure arises because there is a difference between commonly held beliefs about the impact of new automation on human and system performance and the actual impact of new technology on the people who must use it in actual work. The developer tries to justify their new technology in terms of potential benefits, often benefits derived from their <u>assumptions</u> about how the new technology will affect human performance. In contrast, the evidence from research investigations on the actual impact of new technology has shown that many of these assumptions are not tenable.

Table 1 summarizes this contrast by juxtaposing certain beliefs prevalent in developer communities (Column 1: putative benefits or effects of new technology) with the results from investigations of the impact of new technology on human performance (Column 2: the real complexity of how technology affects performance in challenging fields of practice).

Table 1. Designer's eye view of apparent benefits of new automation contrasted with the real experience of operational personnel.

Putative benefit	Real complexity
better results, same system (substitution)	transforms practice, the roles of people change
frees up resources: 1. offloads work	create new kinds of cognitive work, often at the wrong times

frees up resources: 2. focus user attention on the right answer	more threads to track; makes it harder for practitioners to remain aware of and integrate all of the activity and changes around them
less knowledge	new knowledge/skill demands
autonomous machine	team play with people is critical to success
same feedback	new levels and types of feedback are needed to support peoples' new roles
generic flexibility	explosion of features, options and modes create new demands, types of errors, and paths towards failure
reduce human error	both machines and people are fallible; new problems associated with human-machine coordination breakdowns

New user- and practice-oriented design philosophies and concepts are being developed to address deficiencies in human-machine coordination. Their common goal is to provide the basis to design integrated human-machine teams that cooperate and communicate effectively as situations escalate in tempo, demands, and difficulty. Another goal is to help developers identify where problems can arise when new automation projects are considered and therefore help mobilize the design resources to prevent them.

ACKNOWLEDGMENT

The preparation of this manuscript was supported in part under a Cooperative Agreement (NCC 2-592) with the Aerospace Human Factors Research Division of the NASA-Ames Research Center (Technical Monitor: Dr. Everett Palmer).

REFERENCES

Aviation Week and Space Technology (1995). A 330 Crashed In Cat.3 Test Flight. 04/03/95, pp. 72-73.

Aviation Week and Space Technology (1995). A 330 Test Order Included Study of Autopilot Behavior. 04/10/95, p. 60.

Aviation Week and Space Technology (1995). Toulouse A330 Flight Swiftly Turned Critical. 04/17/95, p. 44.

Barber, B. (1983). The Logic and Limits of Trust. New Brunswick, NJ: Rutgers University Press.

Billings, C.E. (1991). Human-Centered Aircraft Automation: A Concept and Guidelines. (NASA Technical Memorandum 103885). Moffett Field, CA: NASA-Ames Research Center.

Billings, C.E. (1996). Aviation Automation: The Search For A Human-Centered Approach. Hillsdale, N.J.: Lawrence Erlbaum Associates.

Carroll, J.M. and Olson, J.R. (1988). Mental Models in Human-Computer Interaction. In M. Helander (Ed.), Handbook of Human-Computer Interaction (pp. 45-65). Elsevier Science Publishers.

Cook, R.I., Potter, S.S., Woods, D.D., and McDonald, J.M. (1991). Evaluating the Human Engineering of Microprocessor-Controlled Operating Room Devices. Journal of Clinical Monitoring, 7, 217-226.

Dornheim, M.A. (1995). Dramatic Incidents Highlight Mode Problems in Cockpits. Aviation Week and Space Technology, (1/30/95), 6-8.

Eldredge, D., Dodd, R.S., and Mangold, S.J. (1991). A Review and Discussion of Flight Management System Incidents Reported to the Aviation Safety Reporting System. (Battelle Report, prepared for the Department of Transportation). Columbus, OH: Volpe National Transportation Systems Center.

Feltovich, P.J., Spiro, R.J., and Coulson, R.L. (1991). Learning, Teaching and Testing for Complex Conceptual Understanding (Technical Report No. 6). Springfield, IL: Southern Illinois University School of Medicine - Conceptual Knowledge Research Project.

Flach, J.M. and Dominguez, C.O. (1995). Use-Centered Design: Integrating the User, Instrument, and Goal. In Ergonomics in Design, July issue.

Gopher, D. (1991). The Skill of Attention Control: Acquisition And Execution of Attention Strategies. In D. Meyer and S. Kornblum (Eds.), Attention and Performance XIV. Hillsdale, NJ: Erlbaum.

Jonides, J. and Yantis, S. (1988). Uniqueness of Abrupt Visual Onset in Capturing Attention. Perception and Psychophysics, 43(4), 346-354.

Jordan, N. (1963). Allocation of Functions Between Man and Machines in Automated Systems. Journal of Applied Psychology, 47(3), 161-165.

Lenorovitz, J.M. (1990). Indian A320 Crash Probe Data Show Crew Improperly Configured Aircraft. Aviation Week and Space Technology, 132 (6/25/90), 84-85.
Leveson, N. G. and Turner, C. S. An investigation of the Therac-25 Accidents. Computer, July, 18-41, 1993.

Moll van Charante, E., Cook, R.I., Woods, D.D., Yue, L., and Howie, M.B. (1992). Human-Computer Interaction in Context: Physician Interaction with Automated Intravenous Controllers in the Heart Room. In H. G. Stassen, editor, Analysis, Design and Evaluation of Man-Machine Systems 1992, Pergamon Press, 1993, p. 263-274.

Moray, N. (1986). Monitoring Behavior and Supervisory Control. In K.R.Boff, L. Kaufman, and J.P. Thomas (Eds.), Handbook of Perception and Human Performance (Vol. 2, Chapter 40). New York: Wiley.

Muir , B. M. (1987). Trust Between Humans and Machines, and the Design of decision aids. International Journal of Man-Machine Studies, 27, 527-539.

Norman, D. A. (1988). The Psychology of Everyday Things. New York, Basic Books.

Norman, D. A. (1990). The 'Problem' with Automation: Inappropriate Feedback and Interaction, not 'Over-Automation'. Philosophical Transactions of the Royal Society of London, B 327, 585-593.

Norman, D. A. (1993). Things That Make Us Smart: Defending Human Attributes in the Age of the Machine. Reading MA: Addison-Wesley.

Rasmussen, J. (1985). Trends in human reliability analysis. *Ergonomics, 28 (8)*, 1185-1196.

Sarter, N.B. (1996). Cockpit automation: From Quantity to Quality, From Individual Pilot to Multiple Agents, In R. Parasuraman and M. Mouloula, editors, *Automation Technology and Human Performance*, Erlbaum.

Sarter, N.B. and Woods, D.D. (1992). Pilot Interaction with Cockpit Automation: Operational Experiences with the Flight Management System. International Journal of Aviation Psychology, 2(4), 303-321.

Sarter, N.B. and Woods, D.D. (1994a). Pilot Interaction with Cockpit Automation II: An Experimental Study of Pilots' Model and Awareness of the Flight Management and Guidance System. International Journal of Aviation Psychology, 4(1), 1-28.

Sarter, N.B. and Woods, D. D. (1994b). Autonomy, Authority, and Observability: The Evolution of Critical Automation Properties and Their Impact on Man-Machine Coordination and Cooperation. Paper presented at the 6th IFAC/IFIP/IFORS/IEA Symposium on Analysis, Design, and Evaluation of Man-Machine Systems. Cambridge, MA, June 1995.

Sarter, N.B. and Woods, D.D. (1995a). "Strong, Silent, and Out-Of-The-Loop:" Properties of Advanced (Cockpit) Automation and Their Impact on Human-Machine Coordination. Cognitive Systems Engineering Laboratory (CSEL) Technical Report No. 95-TR-01.

Sarter, N.B. and Woods, D.D. (1995b). How in the World Did We Ever Get Into That Mode? Mode Error and Awareness in Supervisory Control. Human Factors, 37(1), 5-19.

Sparaco, P. (1994). Human Factors Cited in French A320 Crash. Aviation Week and Space Technology, (1/3/94), 30.

Spiro, R.J., Coulson, R.L., Feltovich, P.J., and Anderson, D.K. (1988). Cognitive Flexibility Theory: Advanced Knowledge Acquisition in Ill-Structured Domains. In the Tenth Annual Conference of the Cognitive Science Society. 17-19 August. Montreal, Canada: LEA.

Wagenaar, W.A. and Keren, G.B. (1986). Does the expert know? The reliability of predictions and confidence ratings of experts. In E. Hollnagel, G. Mancini, and D.D. Woods (Eds.), Intelligent decision support in process environments (pp. 87-103). New York: Springer-Verlag.

Wiener, E.L. and Curry, R.E. (1980). Flight-deck automation: promises and problems. Ergonomics, 23(10), 995-1011.

Wiener, E.L. (1989). Human factors of advanced technology ("glass cockpit") transport aircraft. (NASA Contractor Report No. 177528). Moffett Field, CA: NASA-Ames Research Center.

Wiener, E.L. (1993). Crew Coordination and Training in the Advanced-Technology Cockpit. In E.L. Wiener, B.G. Kanki, and R.L. Helmreich (Eds.), Cockpit Resource Management (pp. 199-223), Academic Press: San Diego.

Winograd, T. and Flores, F. (1986). Understanding computers and cognition. Reading, MA: Addison-Wesley Publishing Company.

Woods, D.D. (1996). Decomposing Automation: Apparent Simplicity, Real Complexity, In R. Parasuraman and M. Mouloula, editors, *Automation Technology and Human Performance*, Erlbaum.

Woods, D.D. (1994). Cognitive Demands and Activities in Dynamic Fault Management: Abduction and Disturbance Management. In N. Stanton, editor, *Human Factors of Alarm Design*, Taylor & Francis, London.

Woods, D.D., Johannesen, L., Cook, R.I., and Sarter, N.B. (1994). Behind Human Error: Cognitive Systems, Computers, and Hindsight (State-of-the-Art Report). Dayton, OH: Crew Systems Ergonomic Information and Analysis Center.