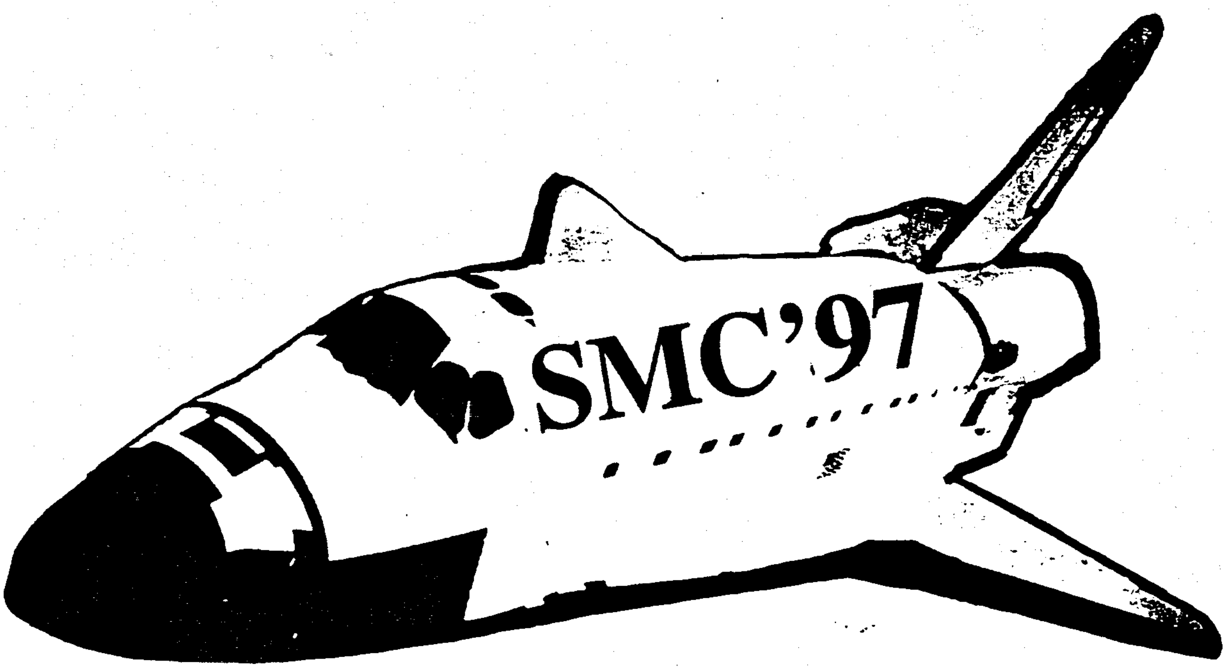
SMC '97 Conference Proceedings

1997 IEEE International Conference on Systems, Man, and Cybernetics

Conference Theme

Computational Cybernetics and Simulation



Volume 3 of 5
97CH36088-5

October 12-15, 1997
Hyatt Orlando
Orlando, Florida, USA

Sponsored by the Systems, Man, and Cybernetics Society of the Institute of Electrical and Electronics Engineers, Inc.

THE 1997 IEEE INTERNATIONAL CONFERENCE ON SYSTEMS, MAN, AND CYBERNETICS
HYATT ORLANDO, ORLANDO FLORIDA, OCTOBER 12 TO 15, 1997
THEME: COMPUTATIONAL CYBERNETICS AND SIMULATION

IMPLEMENTING DECISION SUPPORT TECHNOLOGY AMONG SHOP FLOOR USERS; A STORY OF SUCCESS WITH AN ALTERNATIVE DEPLOYMENT STRATEGY

Lia A. Di Bello Ph. D.
Workplace Technology Research Group
Center for Advanced Studies in Education
City University Graduate School; Suite 620
25 West 43rd Street
New York City, NY 10036
212 642-2543 phone   212 719-2488 fax   ldibello@broadway.gc.cuny.edu

E. Sterling Chamberlain
New York City Transit, Department of Buses
41-36 51st Street, Apt. F3
Woodside, NY 11377
718 507-3638 phone   212 366-1475 fax   esciii@ix.netcom.com

ABSTRACT

This paper explores the recent use of relatively complex decision support systems in public sector industries challenged by privatization and proposes a technology implementation model with proven success. With significant budget reductions, many “public” industries, such as transportation and power generation are finding they must reexamine the way they do business. With the increasing threat of outsourcing, those serious about maintaining their existence are looking more closely at totally new approaches that emphasize the efficient use of funds and labor resources with the goal of maintaining high performance. Many are investigating computerized Decision Support Systems (DSS’s), often borrowing technologies developed for related private industries and tailoring them for new purposes.

Unfortunately, most implementations fall short of expectations. Although acknowledged as potentially valuable—or even as the “only hope”—for saving certain public jobs, DSS’s are considered risky in such environments and most implementations fall far short of expectations. A primary concern is the user; to be effective, DSS’s demand users with 1) an in-depth understanding of their logic and 2) detailed knowledge of he work being done. In public industries such as transit or power generation, mechanics and foremen have the work knowledge needed to be users of DSS’s in their industry, but are unlikely to possess an understanding of computerization and the abstract algorithms driving these technologies. Engineers have been targeted as the end users with mixed success; they are usually not sufficiently close to the work being done to fine tune either system data or variables for analysis.

An iterative implementation model paired with alternative training has proven that DSS’s can be used successfully by the rank and file, with better than planned results. By way of example, we offer the findings of one study of a large public transit property using such a strategy to deploy a preventive maintenance technology. Our study concerns over 1000 shop floor personnel as the primary users and system refiners. A large 24/7 transit property provides an interesting test case because failure of large systems results in significant and obvious declines in service which become quickly apparent.

This paper will detail an iterative Top-down/Bottom-up model for technology deployment consisting of: 1) a relatively small design team of software developers, maintenance managers, shop floor workers and foremen; 2) a powerful new training technique resulting in high quality direct data entry from the shop floor; and 3) feedback from the work floor used in ongoing refinements of the system during roll-out. In addition, we will discuss a multi-level evaluation used to measure the success of the implementation.

The paper concludes that deployment must be organized around the concept that DSS’s are tools that are appropriated by the users and implementation efforts should facilitate that process. This is in striking contrast to “turnkey” approaches. Follow up evaluation data are presented showing how this model leads to more innovative and effective system use over time.

INTRODUCTION

With significant budget reductions, many transit properties are finding they must examine the way they maintain their fleets. With the increasing use of outsourcing to stay within budget, those serious about maintaining their own fleets are looking more closely at totally new approaches that emphasize efficient use of funds and labor resources with the goal of maintaining high fleet performance. While economic functioning was always a goal in maintenance, aggressive strategies were not typically explored in the public transit industry for many reasons, most of which are well known. The first of these reasons is the way that public service budgets are developed and maintained. In the area of transit maintenance they are typically built by “backing into” a total dollar amount which is some version of last year’s budget. This approach ensures that ‘last years’ maintenance will produce ‘this years’ budget. Therefore, there is little incentive for managers to develop and maintain efficient uses of funds; if they come in under budget in one year, their future operating budget may be decreased and the excess shifted to less efficient operations. In other words, innovative managers are not given the option to reinvest their savings and produce a higher quality product, and often see their savings passed on to others.

Secondly, serious disagreement exists over the best ways to reduce costs while providing the most service. In a sense, continuous service and assiduous maintenance are contradictory activities, as maintenance of any kind requires vehicle down time. Most transit properties have tried to manage this conflict by using scheduled inspections, based on various parameters, such as calendar days or mileage. However, this approach identifies parts only after they are visibly defective. At bottom, scheduled inspections of all kinds produce reactive maintenance and can never be truly preventive.

Unfortunately, at this point, nearly all maintenance policy is guided more by politics and folk wisdom than any detailed data collected on vehicles in service. Such practices have seemed necessary – or are accepted as a good second best -- as efforts to collect and maintain analyzable data on large fleets have proven impossible using traditional paper records. However, as transit properties are under increasing pressure to produce strategies with real financial impact, folk wisdom is subject to greater scrutiny.

With advances in information technology, preventive maintenance has been increasingly considered as a potential solution. Many transit industries are investigating computerized Scheduled Maintenance Decision Support Systems (DSS's), often borrowing technologies developed for related industries (e.g. aviation) and tailoring them for transit purposes.

The idea behind Scheduled Maintenance technologies is that, with sufficiently detailed data collection and analysis, the failure rates or life cycles of sub-assemblies and components can be predicted in vehicles exposed to relatively constant conditions (urban vs. rural use for example) and cost-effective preventive replacement of parts can keep vehicles in service. Scheduled part replacement would thus replace other time consuming activities, such as inspections for failure, vehicle breakdowns, and waiting for parts which may not be available for these unplanned needs.

A Scheduled Maintenance system is a very powerful tool for collecting the data required to identify component life cycles and to schedule vehicles for replacement actions. These life cycles can be expressed as the number of days or miles (or other variable) of useful life for each component. A table of life cycles can then be developed and used to create template work orders (work orders that generate themselves according to cycle-based rules). The template work order will then prompt shop floor personnel that a particular part on a particular bus is due to fail and ought to be replaced. Using the tables of life cycles developed with the data collected by a DSS, vehicle maintenance can be planned by coordinating the life cycle driven component replacements. A vehicle can then be scheduled for maintenance at a point that includes several component changes at a time close to the optimum life cycle for each item.

TABLE 1

Optimum component life cycles developed from repair history

	LIFE CYCLE IN DAYS	COMPONENT
1	300	BRAKE RELINE
2	180	AIR DRYER REBUILD
3	280	AIR COMPRESSOR GOVERNOR
4	556	FAN BELT
5	556	FAN BELT TENSIONER
6	260	STARTER RELAY VALVE
7	275	DOOR CONTROL MICRO SWITCHES
8	280	FLASHER UNIT
9	550	DRIVERS SIDE WIPER MOTOR
10	605	FRONT LEVELING VALVES
11	560	DIRECTIONAL FOOT SWITCHES

Unfortunately, most implementations fall short of expectations. Typically two areas are identified in relationship to technology implementation failure. The first is the user; to be effective, DSS's demand users with 1: an in-depth understanding of their logic and 2: detailed knowledge of the work being done. Mechanics and foremen have the work knowledge but are unlikely to possess an understanding of computerization and the abstract algorithms driving these technologies. In addition, many such individuals are resistant to the idea that Decision support technologies are part of their jobs. In contrast, mechanical engineers – the usual target users from the vendor's point of view – often do not possess the in depth knowledge of processes to finely tune these systems [*1*]. The second area of difficulty is the system itself and its distribution or placement in the workplace. Often, "off the shelf" systems are not tailored to work practices or maintenance priorities, and seem irrelevant or awkward to the user. In addition, little thought is usually put into where the technology should go. For example, protection of the terminal or the CPU from theft or dirt often becomes a higher priority than convenient access for the target user.

One system adapted to the transit industry --MIDAS-- has been very successfully implemented at New York Transit's Department of Buses. This paper will detail the three part Top-down/Bottom-up model for technology deployment and the evaluation used to measure its success. As will become clear, this model differs radically from typical technology implementation, and does so, in part, by making the usual cause of technology failure -- the users' resistance surrounding the system -- part of the solution.

THREE PART DEPLOYMENT MODEL

The extensive research that informed the early stages of the MIDAS project also indicated that – for whatever reason – technology implementation practices seemed as limited as the technologies currently in use. For the MIDAS implementation we had three clear goals which forced us to rethink and develop our deployment model. The first of these was a sensitive system that meaningfully represented the work being done on the vehicles. The second was accurate data. Thirdly, and most important, we wanted the system to be actually used; like many properties, we had a history of implementing large technologies that were never accepted or were improperly used. We had even implemented a scheduled maintenance system before that was remembered, by the users, as a legendary waste of money and time. The solution used for MIDAS was to involve the end users in the initial design and to allow users at each site to make small, critical modifications during the actual implementation. This required an iterative approach to the final roll-out.

An iterative design and implementation process.
In the past, new technologies such as a decision support system have been designed and implemented by top management. They tend to have been designed by an IS department or vendor and typically the application design is frozen before it is ever seen by the user. The record of success for systems implemented under this model has not been good. For MIDAS the team decided to use a Top Down/Bottom up model which would encourage changes in the application design during the early phases of the implementation and allow each new site to have input to the way MIDAS functioned in "their" depot. This approach takes seriously the notion that any system that globally impacts work must also map onto what is actually being done, and more importantly, that one may not know the details of such a system without user input. The problem was, how to get user input before MIDAS was in place and overcome the "not invented here" resistance expected in a multiple site implementation? The solution was an iterative design and implementation process. Each aspect of this design is discussed in detail in the following sections.
Briefly, the system was first designed in the basic sense (with the vendor) using a small design team of maintenance and operations staff. Next, an in depth assessment of each site was done by university researchers to measure various aspects of how the depot is functioning and how the workers are understanding and thinking about maintenance operations. This assessment includes cognitive assessments, workflow and artifact analyses and financial performance measures. These data were collected for two purposes: first they allowed the university staff to design a tailor-made educational program for our workers, and secondly, they provided baseline data for later comparison. After the assessment, maintenance workers at all levels were given an aggressive educational intervention (designed using the assessment above) that introduced them to the deeper logic of scheduled maintenance theory and systems. In most locations, these interventions involved as many as 150 managers, supervisors, hourlies and cleaners. The goal was to enable mechanics and line supervisors to do all data entry and analyses. As the training was going on, the hardware and software installation took place. Once all personnel had gone through the workshops, the system went "live" in their location and a "transition team" was placed in the depot until the staff had reached "independence" with MIDAS. Finally, after a number of weeks, the university researchers returned and conducted – again – assessments at various levels to measure change. In addition, they collected user complaints and suggestions as did the transition staff. Design changes were made accordingly and added as enhancements to the MIDAS program.

The users and the deployment were assessed again after several months and further design changes were made. The net result of this approach was a finely tuned system with about 200 user-initiated design changes after being implemented in only two sites. It is expected that the number of changes will decrease dramatically in each new site. It is significant that actual use by educated and involved users identified the finer points of the design at individual locations. This allows personnel at each site to feel that MIDAS was designed by themselves.
Elements of the deployment model

Small, user controlled, design team
In order to tailor the design of the final MIDAS system, the actual end user was included from the very beginning. The design team included experienced operating managers, first line supervision and shop floor mechanics in addition to the usual complement of IS representatives. The group was facilitated by some experienced project managers from the prime contract holder and the developers of the software package we used. The design team met weekly for 4 months and discussed each screen of the application with the developers and outlined how the user would navigate through the system. These designs were applied to the program and put into a test system for further review by the team. This somewhat unusual approach to system design was made more so in that the NYCT design team was chaired by a maintenance manager with full authority to make final decisions. It is vital that end users are in real control of the design process. Information technology department constraints must be justified to the operating users if a conflict is created by design goals. Although a great deal of support is required to educate the operating members this type of small diverse team is the most effective structure for development of a application that fits the end user and avoids major obstacles to acceptance.

An education process that would enable the BMB's to do data entry
At the onset of the design process senior management of the Department of Buses made a decision to have the hourly Bus Maintainers, Class B (BMB's) enter their own repair data into the system without clerical assistance. This was done to ensure that MIDAS recorded bus maintenance history in the most detailed way possible and moved data entry responsibility directly to the shop floor. The idea is that all repairs would be described by the person most familiar with the actual work, the BMB who did it. There is no "interpretation" of an expert mechanics language by a non-expert data entry clerk. On the other hand, although there is much evidence to show that clerical data entry results in significant degradation of the data, there are also numerous examples of the difficulties involved in getting shop floor mechanics to do this type of data entry or to translate their knowledge into the kind of explicit form needed for a computer system.

Since this had never been tried before on such a large scale with a "management" technology, the MIDAS team believed that any traditional education approach based on this very different kind of user would be inappropriate at best. Dr. Lia Di Bello at CUNY Graduate School had established a record at NYCT's car equipment department for overcoming the typical long learning curve and designing educational interventions that seemed to bypass the need for "pre-requisite" knowledge of computers. Based on a program of studies that identified what factors of "on the job" experience led to in depth understanding, Di Bello and her colleagues designed a series of hands-on simulations that basically compress the "incidental" learning that takes place on the job (over a period of a year or two) into a few days. We were also interested in her methods because the workshops have been shown to mitigate resistance. According to DiBello, [*2*] [*3*] [*4*] this may occur because the real basis for resistance may be more cognitive than attitudinal per se. I.e., numerous studies show experienced workers possess expertise that makes it possible for efficient intuitive problem solving [*5*]. Unfortunately the same intuitive obviousness experts experience in their domains may also make it difficult for them to see new possibilities. Di Bello's methods are designed to work with intuitive expertise while re-directing it.

Importantly, rather than pre-packaged workshops Di Bello and her colleagues design simulations that map onto both the historic work practices and the targeted changes in any given site. Based on this record, the implementation team asked Di Bello and Kindred of CUNY to design an intervention for the MIDAS project. They began the process of design with three months of assessment at our sites (see description under “Evaluation” below). After identifying our trouble spots and developing a user profile, they presented us with the workshop design described below.

Description of workshop: The workshop designed by CUNY contains three basic components that have proven to be a powerful combination for effectively learning:

They are hands-on exercises; learners develop new strategies by solving actual problems in miniature versions of their work environments and encounter the actual results of their decisions.

Before being taught new principles, learners are helped (via the activities) to become aware of their "default" strategies and given tools for overcoming/incorporating them into new strategies. The idea here is to get conscious awareness of automatic problem solving strategies and integrate relevant portions with the learning of the new system

Participants learn the logic of new technologies not by operating the computer system, but rather by operating as the computer system, doing its logical operations manually and constructing the necessary data structures. Only after being the computer do users get to use the computer.

For the MIDAS system, CUNY built a three Module workshop. Module One is designed to "engage the default". Rather than anticipate the entry point of learners, CUNY designed a workshop in which participants may construct their own entry. Rather than a lecture, the workshop trainers gives participants a problem to solve with goals that are compatible with the technology --in this case MIDAS-- that they are trying to learn. The trainers also give the participants a number of tools for solving the problem.

For the MIDAS workshop, teams of eight participants were asked to "run a depot" of 40 plastic buses with relatively complex interior components. The goals were to maintain 32 buses in service at all times (limiting the number out of service to eight), order all the materials (within a budget) needed for doing so and evaluate daily operator reports (each “day” being 20 minutes) that might indicate potential problems (e.g., noisy engine). The activity was "rigged" so that the only way to meet these goals was to predict what was due to break next. The breakdown patterns of all components followed time/mileage cycle rules and were pre-calculated using a computer. The participants were given adequate tools to predict and calculate this breakdown, (printouts of every bus’ repair history among other things) but were given other tools as well, including those similar to those used to do "reactive" maintenance.

Close examination of the use the participants make of the tools offered pretty much approximates their work history. People tend to construct a solution to even a novel problem that fits with their experience, even when explicitly instructed to avoid doing so. In fact, the participants are rarely aware they are replicating their normal methods.

Rather than interfere with this tendency, the trainers allow the participants to "wing it", while carefully documenting the cash flow, labor flow, inventory acquisitions and the number and type of on -the -road failures that result from failing to predict problems. Meanwhile, heavy fines are levied for expensive "reactive" problem solving strategies, such as "cannibalizing" an entire bus for a few cheap parts that will get other buses back on the road. Later the participants are shown the consequences of their decision making patterns and the underlying logic used. By the end of the first module, the “depot” is in crisis and the participants are realizing their budget is being expended to react to mounting problems. The activities are stopped and the team is sent to lunch.

In Module Two of the workshop, participants reflect on what they did, as recorded by the trainers. The participants discuss among themselves what thinking led to various decisions and begin to identify practices that lead to bad outcomes vs. practices that are preventative. It is only at this point that the participants are truly open to new ideas about how to solve the problems of vehicle maintenance. They also begin to understand in detail the ways that their “gut feel” decisions reveal how they have actually misunderstood preventative maintenance. In the last part of Module Two, the participants are facilitated by the trainers in building a manual scheduled maintenance system. The participants identify cyclical patterns from histories (which were available from the first but which now take on new meaning) and set up the predictive data structures, identifying true cycles and --most importantly-- coordinating cycles so that their “system” is bringing in a bus only once to satisfy several cycles at once. For example, in the MIDAS workshop, the participants quickly realize that a 15,000 mile cycle and a 30,000 mile cycle can be coordinated so that at least half the time the 15,000 mile cycle co-occurs with a 30,000 mile component cycle. The participants then enter these data on an actual test region in MIDAS and create and assign the work-orders according to this schedule.

During Module Three, the participants complete their data entry and print out their work assignment sheets and work orders. They run their miniature depot again using MIDAS and see the difference in profits and ease of workflow. Usually only after 5 “days” the team can afford to buy an additional bus to add to the fleet and thereby increase their farebox revenue.

The last activity of the workshop involves entering the data on work-orders (paying attention to detailing the components, defects and symptoms involved) and closing out both work orders and work assignment sheets. At this point, participants also learn how to get various reports that they now realize they will want, such as a 30 day history on a bus.. After operating *as* MIDAS and then *with* MIDAS, participants navigate through the actual system more easily, know what to look for and ask informed questions. Even computer illiterate individuals show little hesitation when exploring the system.

Di Bello and Kindred believe that the workshop’s power lies in the first two Modules, when the participants actively encounter the basis for their resistance --existing expertise and automatic ways of doing things-- and re-tailor it to fit new demands and priorities [*6*]. The process of doing this is critical because each individual has a different prior perspective that must be reckoned with.

The research informing this method has shown it is best to facilitate this process of “re-tooling” one’s way of thinking rather than try to overlay new knowledge on top of the old with lecturing. It seems that learning always involves building upon, or reorganizing the way that one already understands something [*1*]. Therefore, it is critical to engage prior knowledge, if only to eventually make sure it is changed or reorganized. Why is it necessary to let the participants do this on their own instead of asking trainers to instruct differently? Research shows that learners of diverse work history backgrounds "enter" into understanding by multiple [*2*]. For example, whenever a teacher "simplifies" material for his or her students, he or she is really anticipating the "entry point" of the learners. This method often fails with experienced workers because the entry point is not always predictable or universal (for example, "simplified" is often not helpful for those experienced in thinking through vast amounts of detail). The learners --when allowed-- actually do better at “breaking it down for themselves” in a way that is useful to them.

Modifying MIDAS from user suggestions and requests

From previous experience and from discussions with other transit agencies the MIDAS team recognized that implementing a “final” design of the MIDAS system would be very difficult for two reasons. First, it is difficult to fine tune a systems content before it is in use, on-site. Second, having 23 separate and diverse maintenance shops and depots involved introduces some inherent difficulties; each new location would react to MIDAS with “it won’t work here” and would feel that the unique qualities of ‘their’ shop had not been accounted for. As a solution the MIDAS team decided to seek out suggestions and requests from the users at each location as the MIDAS system was implemented and to add these enhancements before the transition was complete. The technical structure to support this level of change must be available in the basic application and is a very important consideration in the selection of a system. In MIDAS most processes are controlled by tables which are maintained by the user from within the MIDAS program. User requests for a different maintenance cycle, a more specific description or a new repair or component can be easily met by a transition support member at the location. A new report or a search requires some programming, either by the developers or in-house staff, but are added in most cases as requested. The relationship with the vendor must allow for such programming changes.

THE EVALUATION

In addition to the training, Di Bello and Kindred collaborated with New York City Transit on the design of a three level evaluation to measure change and to identify “trouble” spots while the MIDAS implementation was in progress. The three levels comprised:

1. Cognitive probes
2. Analysis of usage patterns using login records and keystroke recordings
3. Analysis of financial impact as measured by Mean Distance Between Failure (MDBF is the number of in service vehicle failures divided by the in service distance traveled).

A detailed discussion of each of these methods is beyond the scope of this paper but a brief description of these assessments will render the results more comprehensible.

First, the trainees were tested on their understanding of scheduled maintenance concepts using a specially designed cognitive battery. The battery was a set of tasks constructed to get at implicit, spontaneous strategies, and not simply memorized verbal material. This method of getting at implicit knowledge in a target domain was developed in previous research on DSS’s and proven highly sensitive for getting at in depth knowledge gained from experience. It was adapted to this project by re-designing the battery to target “reactive” and “proactive” ways of analyzing machine breakdown [*3*]. The “testing” was conducted in one-on-one interviews with volunteer mechanics. Secondly, usage logs were downloaded from the MIDAS system and analyzed for complexity and accuracy of data input. For this analysis, data input by mechanics who participated in CUNY workshops were compared with a control group (mechanics who were vendor trained or trained on the job, but who had equal work-time experience using MIDAS). The first of these logs was collected 7 months after MIDAS went “live” and continuously thereafter. Third, the depot’s Mean Distance Between Failures (MDBF, the number of in service vehicle failures divided by the in service vehicle mileage) for the months before and after MIDAS implementation were compared with that of other depots where MIDAS was not yet implemented.

TABLE 2

Cognitive Battery Results after six months; Proportional Scores on the “History” tasks, scores shown by sub-task. Note: both training groups were identical to the control group before training. Post test scores show a marked decrease in “traditional” reactive maintenance strategies and a marked in crease in proactive “Scheduled maintenance” strategies after training. The effect was strongest with those who underwent two days of training. This effect transferred across objects.

“Trad.” indicates proportion of all possible traditional strategies performed;
“SM” indicates proportion of all possible proactive preventive planning strategies performed.

	Real Bus		**Toy Bus**		**Machine T**	
	Trad.[a]	**S.M.[b]**	**Trad.[c]**	**S.M.[d]**	**Trad.**	**S.M.**
Control Group	**.58**	**.39**	**.63**	**.38**	**.43**	**.32**
One-Day Work Shop	**.42**	**.58**	**.43**	**.51**	**.34**	**.48**
Two-Day Work Shop	**.38**	**.62**	**.37**	**.65**	**.35**	**.60**

[a] ANOVA indicated significant differences between controls and 2 Day Trainees; $F(2,27) = 5.70$ $p< .008$
[b] ANOVA indicated significant differences between controls and 2 Day Trainees; $F(2,27) = 4.32$ $p< .02$
[c] ANOVA indicated significant differences between controls and 2 Day Trainees; $F(2,27) = 4.48$ $p< .02$
[d] ANOVA indicated significant differences between controls and 2 Day Trainees; $F(2,27) = 6.68$ $p< .005$

The pilot project was judged largely successful. Follow-up, one-on-one testing, was done six months after training. Mechanics who attended the workshops had a much deeper understanding of MIDAS and scheduled maintenance concepts and were transferring this knowledge to their thinking about the buses they worked. In striking contrast, the control group continued to conceptualize defects in repairs within a traditionally reactive paradigm.

TABLE 3

Analysis of data entry patterns; QV mechanics during the month of April 1995 [a].

Average frequency of using a particular component code in response to bus-driver determined symptom codes; lower mean indicates greater variance of response and more detailed and accurate data. Small standard deviation indicates a homogeneity effect.

	mean	standard deviation
Control group [b]	1.70	1.07
1-Day Training	1.55	.52
2-Day Training	1.34	.27

[a] System went "live" October 1994.
[b] Control group was vendor trained and had been using the system on the job as long as the workshop trained groups.

Downloads of workers' navigation through the system and data entry practices also showed that trainees entered component data that was more than twice as detailed as the control group. There was also considerable within group consistency in the type of detail entered; i.e., statistical analysis showed little within group variation even though the input was much more complex. In contrast, the control group entered much less detailed component information and showed significantly greater idiosyncrasy in their choices of what to enter.

TABLE 4

Users choice of component codes used to identify problems in vehicles six months after workshops and system implementation. Numbers below indicate the number of different codes used by each group at each level of database detail. High numbers of codes in the higher levels indicate more detailed data entry.

general <-------------------------->specific

Level	1	2	3	4	5	6
Control Group [a]	10	24	48	57	16	9
Trainee Group	10	33	62	126	25	15

[a] control group was vendor trained and had been using MIDAS as long as the workshop trained group

Lastly, MDBF (Mead Distance Between Failures) for the pilot depot was higher overall, but more importantly, showed a gradual incline while non-MIDAS depots, and New York City Transit as a whole, were experiencing a decline.

CHART 1

MIDAS Vs Non MIDAS Depots: MDBF

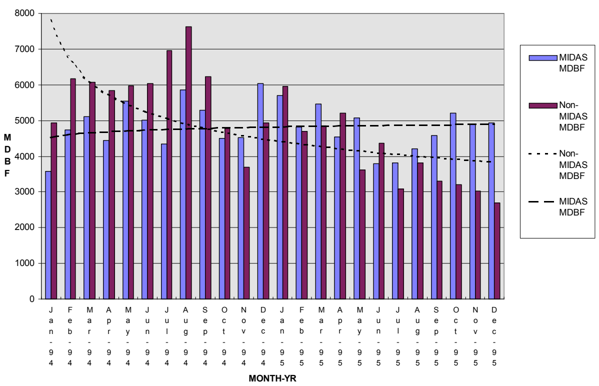
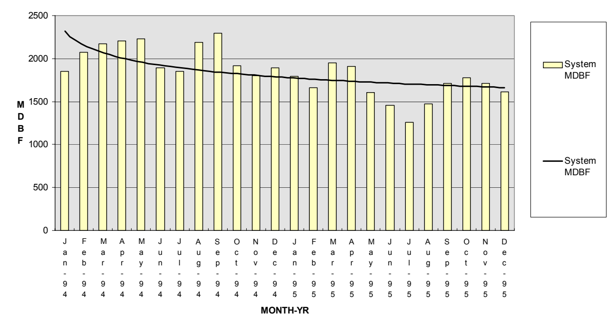


CHART 2
System-Wide MDBF



Qualitative data supported the general trend. For example, over 200 design changes in the system were worker-initiated after the pilot. In addition, workers' complaints about the system and its deployment show a sophisticated understanding; they wanted more terminals close to their work benches and asked that the data tables reflect more detail so that they could better describe their work. Vandalism of terminals left in open work areas --a concern of upper management-- has not occurred at all.

CONCLUSION

There have been many reasons invoked for the frequent failure of technology implementations and most acknowledge that the "user" is a significant factor. In this paper we have sought to show that involving the user can actually increase the chances of success and give some stability and depth to the technology's overall impact. However, as should also be clear, "considering the user" is not a simple matter of a few presentations, conducting surveys, or even a matter of giving traditional training courses. Rather, this paper proposes an iterative implementation process that incorporates alternative models of worker education, project evaluation and ongoing application modification on site. We consider these components to be key issues and yet not easy to address. On the other hand, the payoff is well worth the extra work. For example, we estimate that the educational and design modification processes comprised at most 10% of the total project budget (this includes the cost of overtime used to replace workers being trained) and yet drove its chances for success from 40% to 90%. Since we are already seeing an financial payoff, (See Charts 1 and 2) this seems a small price to pay.

REFERENCES CITED:

[1] Di Bello, L. & Spender J-C (1996). Constructive learning: a new approach to deploying technological systems into the workplace. *International Journal of Technology Management.*

[2] Di Bello, L. (1996). Providing multiple "ways in" to expertise for learners with different backgrounds; when it works and what it suggests about adult cognitive development. Special Issue of the *Journal of Experimental and Theoretical Artificial Intelligence*.

[3] Di Bello, L. (1996). Exploring the relationship between activity and the development of expertise; paradigm shifts and decision defaults. In *Naturalistic Decision Making*, (C. Zsambok & G. Klein eds.) ".

[4] Di Bello, L (1996) *Education for complex workplace technology; unpacking the assumptions of training that fails.* Invited presentation for the Second Workplace Technology Conference, Russell Sage Foundation, New York NY. July 31. (Submitted for Publication)

[5] Dreyfus, H. (1997). Intuitive, Deliberative, and Calculative Models of Expert Performance. In Naturalistic decision making, Zsambok and Klein (eds.) Erlbaum NJ.

[6] Kindred J. & DiBello, L. (1996). *Reframing in depth learning and development: Comparing constructive activities and knowledge "replacement" with traditional training and knowledge "acquisition".* Presented as part of Structured Poster Session "Sociocultural studies of development and learning at work", Annual Meeting of the A.E.R.A. April 22, Hilton Hotel, San Francisco CA. (Submitted for Publication)