

1995128357

N95-34778

Is There A Role For A "Test Controller" In The Development Of New ATC Equipment?

Ron Westrum

Eastern Michigan University

Introduction

In the aviation field, test pilots have long performed a valuable function in the evaluation and improvement of new aircraft. Through their special experience and training, test pilots are able to provide expert feedback for the development process. Although often glamorized by films and books, the role of the test pilot is basically that of a member of the engineering team. The test pilot checks out the plane in the air, explores the performance envelope of the various aircraft systems, notes "bugs" and other infelicities of the equipment, and makes suggestions for improvements. Test pilots have unusual piloting skills, but more importantly have training in systematic check-out and a high sensitivity to performance quirks that others might miss (Hallion, 1981).

Testing new ATC equipment necessarily involves similar skills to test-piloting. Check-out is expected to take place according to systematic protocols, and problems in operation are expected to be spotted and removed. Who is qualified to do this? And what impact will the involvement of controllers at various stages of the development process have on the effectiveness of equipment finally released? Patrick Dujardin (1993) has suggested that early involvement of controllers in the R & D process may discourage important advances, since controllers will feel comfortable only with equipment that seems familiar. There is broad agreement, in fact, that early involvement of working controllers is likely to lead to compromises or kludge designs. The regular controller is unlikely to want to "push the envelope." Many observers have remarked that equipment used by the FAA, both currently and in the near future, reflects this conservative attitude.

A test controller, however, would not share the same bias against new equipment. Note that this is a different role from the evaluation of finished systems. The "test controller" would be used to seeing equipment in raw form, just as a test pilot would be. Again following the analogy with airplane development, a test controller would have to be recognized as a top practitioner, with the respect of other controllers. Such a person's certification of the equipment to the working controller, then, would be one guarantee that the equipment, if not trouble-free, would be at least safe and efficient to use.

New hardware and software now face stiff resistance if they originate from someone other than facility automation specialists. For instance, FAA-produced software for airport ATC systems has many credibility problems. It is not perfect, and in any case needs to be customized

Human Factors Certification of Advanced Aviation Technologies
Edited by J. A. Wise, V. D. Hopkin, and D. J. Garland
Copyright © 1994 Embry-Riddle Aeronautical University Press

PRECEDING PAGE BLANK NOT FILMED

PAGE 220

to handle site-specific problems. Currently, having capable controllers is the best guarantee against software's inadequacies. Most controllers use "so-so software developed by someone" as Jim Schmidt of Martin Marietta puts it. Controllers must carry on from where the software leaves off, bridging between it and the operating situation. Potentially, certification by a "test controller" that new equipment satisfies human factors requirements might give controllers the confidence they need to master complex new equipments and procedures.

Another important feature of the test controller is checking out the "far corners of the envelope." In the R & D process, early efforts are focused on getting the system to work. But test controllers need to try to make it fail, to exercise it, as it were, beyond ordinary limits, to eliminate the hidden bugs. Ideally, of course, a better process would be developed for getting error-free software. In real life, however, automated systems are likely to possess "glitches" difficult to eliminate. For instance, a recent *Wall Street Journal* article reported that the Honeywell autopilot installed in Boeing 747's behaved in mysterious ways. The FAA noted about 30 incidents, including a recent near-crash over Thunder Bay, Canada, involving malfunctioning autopilots. Experts have been unable to isolate the fault (Carley, 1993). Thus, test controllers need to check the system out using "impolite" actions. This is very similar to what Sir Karl Popper recommends in his book *The Logic of Scientific Discovery*: propound bold hypotheses, and then give them severe tests (Popper, 1961).

Before we go further, however, we need to consider the innovation process itself, since a test controller will have to fit into it. Innovation in the United States Federal Aviation Administration is a very problematic process. We need to examine it in a bit more detail before going further.

Innovation: A Long Haul

Charles Franklin "Boss" Kettering once said that "getting a new idea into a factory is the greatest durability contest in the world." He might have said this about Air Traffic Control. The current process by which new ATC equipment is introduced is slow and inefficient in many ways.

- 1) There are excessive delays, on the order of a decade or more, from the time new equipment is developed until it is actually used. Thus, by the time the equipment is installed, it has usually been obsolete for years. It may nonetheless represent a real advance over what was used before. Many airports function with equipment that controllers think properly belongs in museums. This is demoralizing both to innovators and operators.
- 2) An incremental approach is used. This approach forces new equipment to be compatible with current equipment, often leading in the end to an inelegant, "kludge" design, rather than an optimal re-design from the ground up. While each new piece of equipment may function well on its own, nothing guarantees either its compatibility or lack of redundancy with current equipment.
- 3) Use of political fiat is sometimes used to impose "quick fixes" that need better testing before implementation takes place. In many cases these programs fail to work as planned and thus increase barriers to further innovation. Problems include failure to introduce new equipment effectively, to take learning curve considerations into account, appropriate check-out by "test controllers," and well-conceived instructional methods.

4) Very high stakes are involved in securing government contracts, leading to intense struggles on the part of private firms to get their product accepted. Because competing parties often resort to legal action to block or reverse decisions already made, delay is common. As with defense contracting, the long haul involved and the "winner-take-all" outcomes often result in selection of contractors who are good at lasting through the many rounds necessary to win a contract; these are not necessarily the contractors with the best systems.

The current system is designed to include three parties: private firms that do the actual hardware development; FAA higher officials, who make decisions about which devices to install; and controllers, who will actually use the devices, once they are officially accepted. In principle, controllers develop needs, these needs get expressed as FAA requirements, and private industry responds to the requirements by hardware or software innovation. But there is a built-in paradox. The paradox is that controllers do not know what they can ask for until they know what can be developed. Vendors, on the other hand, often do not understand what controllers need. FAA higher authority, trying to bridge the gap between needs and products, is hemmed in on one side by political and legal constraints and on the others by vendors jockeying for contracts and FAA facilities fearful of clumsy automation.

Something of the complexity of bringing a new system on line is revealed by the failure of the IBM® Advanced Automation System to be implemented on schedule (Burgess 1993). The Advanced Automation System (AAS) will cost something over \$4 billion to develop in a joint effort between IBM and the FAA. It will replace the current generation of mainframe-generated pictures for controller positions with personal computers, and will offer much more flexibility. But the FAA continually revised the specifications, and IBM seems to have created its own delays. The system will require something like 1.6 million lines of code, about the same order as "Star Wars." The specification documents themselves would form a stack about three and a half feet high. In basic terms, the contract involves the normal delays and overruns of the typical big-ticket military weapons system project. This is not a good sign.

Since the system has long delays, attempts to get around the normal channels are common. As Lee Paul (1979) has written, "The number of years required by an orderly development process results in irresistible pressures to bypass the system." This often leads to ill-considered moves, including "designs by fiat" that not only fail but also prejudice future attempts at innovation. The formal system also largely ignores the automation specialists (see below) and other members of the system who often have excellent ideas, but who are not considered partners in the innovation process. On the other hand, there has been long-term involvement of controllers on work teams.

These complex dynamics do not bode well for getting the right control equipment to the right people in the right time frame. A top priority for the FAA might well be to examine its own innovation process,

Patching It Up

Ironically, the controllers often seem to do better themselves through informal networking when it comes to customizing ATC software. While some software can be originated locally, hardware on the other hand necessarily is produced off-site and centrally tested and introduced. Still, because each site has slightly different requirements, software can be generic only up to a certain point. Beyond this point, software must be customized for the specific site. This is done

through "patches": software oriented to site-specific problems, and written by local "automation specialists." For instance, at Detroit Metropolitan Airport (DTM) the FAA-produced A-305 ARTS-III software (introduced in 1993) was voluminous. Documentation for the software was four large volumes, each a folio volume the size of a desk encyclopedia. The cross-reference book alone weighed 20 pounds. Yet the software required 190 patches to adapt it to local conditions. The automation specialists on the staff estimated that development of this supplemental software required several months to complete, exclusive of already existing site-specific software, which also had to be changed. In principle, software received from the FAA is supposed to be implemented "as is." The reality is that this cannot be.

At Chicago O'Hare Airport, for instance, there was at one time a rule that strictly limited the number of local patches. This rule, however, did not make sense, and so was constantly bypassed. Each time local controllers asked for a specific change, the patch would be added to an existing patch, which clearly flouted the spirit of the rules, though superficially legal. This bypassing of the system was never formally acknowledged. Nonetheless, the local programmers thought FAA officials must have been aware of it.

While sites are often different, many sites share the same problems. The obvious thing, then, is to make sure that a site has available to it any patch in the system that will solve its problems. Although all patches used anywhere are included on a list sent to all "automation specialists," this list is seldom seen by controllers, and is hard to interpret in any case.

There are about 200 automation specialists in the United States. They are former controllers now responsible for software management at the control centers. They are expected to act as the local interface between the needs of controllers on one hand and the provision of new automation through borrowing patches or getting local technicians to do the programming. However, they usually have their hands full with programming responsive to demands by the local controllers for various kinds of minor fixes. One major airport had a list of 30 such patches waiting to be programmed; this is fairly typical. These demands are often either made by top management or presented through union channels, which makes them hard to ignore. Useful patches, then, may often get lost in the system's complexities because they are not available in a user-friendly way.

The automation specialists have more credibility because they are former controllers. Knowing the job that the controller must do in some detail makes their products far more user-friendly than it might otherwise be. But few have college degrees in computer science. Some do not have college degrees at all. They get considerable in-service training from the FAA, but both they and others believe that their programming would be superior if they had more computer training.

No one knows how much of the innovation in the system is actually due to the local automation specialists. For instance, a program called Cenrap allows the local facilities to get radar screen pictures even if their antenna goes out, by getting information on plane positions from more powerful Regional Center (ARTCC) radars. This program was reportedly suggested in about 1985 by either an automation specialist or a technician who realized that capabilities already in use, with a little extra work, could provide back-up radar pictures for facilities that lost their radar but not their system (ARTS-III) software.

Local Content

To compensate for the system's inadequacies, informal networking often provides the primary channel for patches to travel from one center to another. In the FAA southern region, for

instance, an informal computer bulletin board provides information about patches. Other information comes about through individuals who move from one site to another, or who through union duties or curiosity circulate through the system. Actually watching a patch in operation may be more valuable than reading an abstract about it.

Yet informal networking clearly is a second-best to a user-friendly system for spreading information about patches. Why is there not a dedicated "patch specialist" who knows what is available who travels through the various sites?

Similarly, site development of patches is often done partly *sub rosa*. Legally, patches must be run through Washington D.C. for regulatory approval before they are used. The formal approval process takes about a year. After approval, the patch is sent back to the site for on-line testing. But local automation specialists do not want to send a patch through the system until they know it works. And how do they know it works? They try it out. To try it out, they need a computer. Often the only computer available for the purpose is the Center's main computer. It would be best to try the patch out on a mock-up computer off-line, but mock-up computers can cost as much as the main computer. So the patch is run on the main computer at a lull time, such as 2 AM. Controllers will almost never try to control aircraft with experimental software. Locking up the system would be both dangerous and would jeopardize their jobs. But without a realistic (i.e., live) test, they do not want to release the software. So while planes are controlled through some other method, the patch is tried out. Once it is known to work, it is sent through the formal process.

[I was unable to gain any information regarding site-generated patches for the regional (en route) centers. Ostensibly, all patches for regional centers must originate from the FAA Technical Center in Atlantic City. To proceed otherwise risks severe legal sanctions.]

Higher echelons of the FAA must know that this kind of covert experimental activity goes on, although they cannot publicly either acknowledge or condone it. However, while obviously better than a paper check-out of the software, this "skunk works" approach has some dangers. One of the problems is that fewer programs result from it than would if it were openly acknowledged. Controllers and automation specialists would both get in serious trouble if they were caught operating with an illegal patch. It would be better if the test were carried out openly. But the best would be creation of what Lee Paul calls a "more forgiving environment," where experiments with patches could be run off-line, a full-scale simulation facility that could be customized temporarily to run a Center's software.

Controllers' experience with innovation has largely been negative. Good ideas by those lower down in the system often seem to get stone-walled or put on the back burner. Ideas that come down from the top are often half-baked or flawed. But the strongest message about innovation is the equipment with which controllers in the U.S.A. are forced to use. State-of-the-art aircraft are controlled by ATC equipment which is often two or three generations out of date. Whether the explanation for this state of affairs is politics, bureaucracy, or sheer conservatism, the message it sends is one of stagnation and indifference. "Good enough for government work" seems to be the limit the controller can expect. Controllers have to be good; their equipment is not.

Test Controllers

The innovation process for new hardware and software occurs along a timeline that can largely be considered in three phases: research and development, preliminary testing, and full-scale deployment. There is a role for controllers in testing new hardware and software in each of these phases.

Research and Development. A role in R & D means that the controller would act in the same role as a test pilot. He or she would encounter new equipment in its formative stages and would be able to help suggest improvements which would move the system from the prototype to the operational stage. The FAA currently is using controllers on its work teams for the new consoles that IBM is developing in Rockville. Some of these controllers have been on the teams for ten years. However, unlike the system for test pilots, there is no way that the experience of these controllers as test controllers is recorded other than in the design of the equipment. They are simply sent back to their centers after they finish their tasks. Note also that at many ATC facilities, local software will be tested by individuals who serve the role of test controller for that facility, even though the term as such may not be used. Donald Pate at the Standards Development Branch of the FAA in Oklahoma City similarly uses journeymen controllers for his experimentation and standard-setting.

One anonymous observer pointed to a problem with the use of ordinary controllers in the R & D process. Early involvement of routine users tends to lower the team's sights, and thus may lead to incremental changes rather than re-design from the ground up. An example is the FAA's use of a "Sector Sweep Validation Team" involving ordinary controllers early on in the process. Ultimately, the console produced by the team was a kludge design, according to this individual. Thus early involvement can lead to dangerous compromises. A "test controller," in principle at least, would have enough experience with the innovation process to be less bothered by radical innovation.

A second problem to which union representative Larry Barbour called attention is the attrition of skill among controllers who are promoted to supervisor. Several individuals noted that supervisors could no longer be considered proficient in acting as controllers, once promoted. During the PATCO strike, many of the supervisors actually had to do some controlling. This was a frightening experience for some of them who had lost their skills. "I remember watching some of these guys with sweat pouring down their backs," said one observer. Yet several of the IBM-design work teams contained a majority of supervisors by the time the project was finished. One work team had one controller and eleven supervisors! This, however, would be a problem for test controllers, too. Some method for alternating actual control experience and innovation activities would be necessary.

Preliminary Testing. In this phase, the overall design of the software or equipment is fixed, and the purpose of testing is to eliminate any remaining "bugs". The role of controllers here is to act as intelligent customers rather than test pilots per se. It is often during this phase also that software can be customized for a particular site. Hugh Bergeron and Harold Heinrichs report their experiences in using controller "cadres" first to test software, and second to act as trainers, both at Denver and at Dallas/Fort Worth. These experiments seem to have been very successful, though the system was not completely ready for them (Bergeron and Heinrichs, 1993).

Bergeron and Heinrich's cadres might be seen as somewhat analogous to the New Equipment Introduction Details (N.E.I.D.) used by the U.S. Signal Corps in World War II. The Signal Corps, finding that newly developed equipment typically was not accepted in the field, developed a kind of special detachment under the leadership of a Lt. Col. Jensen. The detachment included no one without a uniform, and no one ranking above a major or below a sergeant. It always accompanied the equipment from the point of origin (the factory) to the field with no hand-offs. The Signal Corps discovered that this scheme was so successful that it could not get equipment into the field without an N.E.I.D. Unfortunately, the value of this device was not recognized after the war, and so no one studied it. Its only mention in the official history of the Signal Corps in World War II is a tiny footnote.

Full Scale Deployment. During this phase, controllers are still important as intelligent users. Cadres who have been used in phase 2 to work out bugs can act as brokers between laboratory and ATC facilities to transmit information backwards and forwards between designers and users. As equipment and software is given a full-scale test, limitations and bugs will become apparent. Often this may mean moving the novel hard- or software to a new location, with new demands. IBM, using Seattle Regional Center (ARTCC) as a test site, is rumored to have eliminated phone jacks from the controllers' positions which a "D man," used extensively in the busier centers to work computers and assemble flight strips, could use. Protests by Cleveland Center (and others) quickly got the jacks back. Developing effective channels for user feedback is thus very important.

Discussion And Conclusion

Earl Wiener points out that human factors problems fixed during the R & D stage are paid for once. When they are not fixed during R & D, they are then paid for every day. How users are involved in the R & D process to assist in developing equipment is a critical issue. Effective involvement can produce real improvements. Ineffective involvement can produce inefficient kludges or systems that are actually dangerous.

The underlying problem is the management of information and ideas. To develop a really generative system (see Westrum 1993) a great deal would have to change in the way that the FAA innovates. Use of test controllers would solve only some of the problems. For instance, we have cockpit resource management now for pilots; we may have it soon for controllers. But the management of ideas in the innovation process also needs intellectual resource management. Simply involving users is not enough. Brought in at the wrong point in the development process, users can block or compromise innovation. User involvement must be carefully considered. A test controller may be one solution to this problem. It might be necessary to have several kinds of test controllers (en route versus TRACON, for instance). No doubt further problems would surface in getting test controllers into operation.

I would recommend that the FAA engage in a series of case studies of controller involvement in the innovation process. A systematic comparison of effective and ineffective cases would do much to clarify what we ought to do in the future. Unfortunately, I have been unable to find any cases where test controllers have been used. Perhaps we need to create some, to see how they work!

Acknowledgement

Many individuals took considerable time and effort to assist the author in writing this paper. These persons include Larry Barbour, Alex Becker, John Dill, Dick Edsall, Ron Leoni, Don Pate, Lee Paul, Willis Richardson, Jim Schmidt, Steve Walter, Marty White, and above all, Dud Tenney. While these persons contributed, the opinions expressed in this paper are solely those of the author and do not necessarily reflect either the views of the meeting's sponsors or of any individual who assisted in the paper's preparation.

References

- Bergeron, H., & Heinrichs, H. (1993). A training approach for highly automated ATC systems. *Seventh International Symposium on Aviation Psychology*, Columbus, Ohio.
- Burgess, J. (1993). Out of control contract. *Washington Post*, Washington Business Section, March 8, pp. 1, 22, 23.
- Carley, W. M. (1993). Jet's near-crash shows 747's may be at risk of autopilot failure. *Wall Street Journal*, April 26: A1, A6.
- Dujardin, P. (1993) The inclusion of future users in the design and evaluation process. *Le Transpondeur*, April, pp. 36-39.
- Hallion, R. P. (1981). *Test pilots: The frontiersmen of flight*. Garden City, New York: Doubleday.
- Paul, L. (1979). How can we learn from our mistakes if we never admit we make any? *Proceedings, 29th Air Traffic Control Association Conference*, Fall, pp. 194-198.
- Popper, K. (1961). *The logic of scientific discovery*. New York: Science Editions.
- Westrum, R. (1993). Cultures with requisite imagination. In J. A. Wise, V. D. Hopkin, and P. Stager (Eds.), *Verification and Validation of Complex Systems: Human Factors Issues*. New York: Springer-Verlag.