

# MANAGING ADA PROJECTS: THE PEOPLE ISSUES

presented at

TRI-Ada '88  
Charleston, WV  
26 October 1988

by

DONALD G. FIRESMITH



President  
Advanced Software Technology Specialists  
3418 Broadway  
Fort Wayne, IN 46807  
(219) 456-9260

Donald G. Firesmith is President of Advanced Software Technology Specialists, an Ada-oriented software consulting and training company. He is also founding Chairman of the ACM SIGAda Software Development Standards and Ada Working Group (SDSAWG) and its Joint Integrated Avionics (JIAWG) Support Subgroup (JSS). He is a representative of the Electronic Industries Association (EIA) on the Council of Defense and Space Industries Associations (CODSIA) Software Development Standards Task Group. He has been influential in obtaining many Ada-oriented improvements in DOD-STD-2167A. While at Magnavox, he developed the Object-Oriented Development (OOD) method used on the Advanced Field Artillery Tactical Data Systems (AFATDS) program. He has since developed the Ada Development Method (ADM), an object-oriented software development method. His Ada Project Management seminar has been presented over 25 times in the United States, Europe, and Australia.

# Managing Ada Projects: the People Issues

Donald G. Firesmith, President  
 ADVANCED SOFTWARE TECHNOLOGY SPECIALISTS  
 3418 Broadway  
 Fort Wayne, IN 46807  
 (219) 456-9260  
 firesmith@ajpo.sei.cmu.edu

August 4, 1988

## Abstract

In order to successfully manage the new Ada technology, the software manager must be able to: successfully interact with upper management, negotiate with and educate the contracting agency, user, and Independent Verification and Validation (IV&V) personnel, and lead an often diverse group of developers using new software development methods and tools in addition to a new language. This paper points out problems and offers solutions for dealing with project politics, sociology, and psychology that often have a larger impact on the success of Ada projects than do the technical issues.

## Introduction

As Machiavelli once noted, "There is nothing more difficult to take in hand, more perilous to conduct, or more uncertain in its success, than to take the lead in the introduction of a new order of things." Ada is such a technological transition, and the success of Ada projects often depends as much (or more) on the software manager's grasp and control of social, psychological, and political forces as it does on the manager's understanding of the important technical issues.

The software manager must be able to: successfully deal with upper management in order to obtain the necessary funding for training, tools, and hardware; schedule modifications due to new life-cycle models and the new emphasis on requirements analysis and design; and support for (and understanding of) the innovative approaches associated with Ada. Similarly, the manager must be able to negotiate with and educate the contracting agency, user, and Independent Verification and Validation (IV&V) personnel. Finally, the manager must also lead an often diverse

group of developers using new software development methods and tools in addition to a new language.

## Upper Management

Although upper management Ada savvy and support is critical to the success of each corporation's Ada technology transition (as well as the success of its individual Ada projects), both corporate and project management typically exhibit a very disturbing lack of awareness of the important impacts of Ada. Ada and Ada-oriented software development methods are usually new to them and are often looked on with suspicion by both technical and administrative managers whose past experience with other languages has suddenly become less relevant. The Ada transition is as significant as was the transition from vacuum tubes to solid state, and managers should clearly point out to upper management the dangers of hesitating (or resisting) during such a technology transition, especially when the customer is demanding the change. The manager should also mention the advances and investments made by the competition (e.g., the Unisys Defense Systems \$10 million Ada Program Office and the General Electric Ada Training Lab).

Because upper management is often more oriented towards and receptive to financial arguments, the software manager should use Ada cost models such as SOFTCOST-ADA<sup>(tm)</sup> and provide a cost/benefit analysis. Practical information should be presented in terms of market size and projected market share, capital investment, financial impact, personnel needs, the corporate way of developing software, the risks, and means of managing those risks, etc. Above all else, Ada should not be oversold. Because of past over-optimistic promises prior to the recent proliferation of production quality compilers, initial pro-

jections of increased productivity may well need to be artificially low (e.g., 25%) to be believable. If rework is currently a major source of income (one that will decrease as Ada increases the quality of delivered code), the manager may need to show the projected impact of the current trend towards warranties and fixed-price contracts. The manager must also modify upper management's expectations regarding project phasing, milestones, manpower loading, etc.

Finally, because people often are more motivated for personal and emotional reasons than they are by technical arguments, the manager should determine and discuss personal benefits with individual executives and how they may mitigate Ada's risks to their personal careers.

## Contracting Agency Personnel

Contracting agency, user, and IV&V contractor understanding and support for Ada is also critical to project success. Yet contracting agency, user, and IV&V contractor personnel typically exhibit an even more disturbing lack of awareness of important Ada impacts than upper management. As Dr. Yale Jay Lubkin, Electronic Warfare Editor of *Defense Science and Electronics* noted, "With rare exceptions, the kind of people who procure and administer contracts are not the kind of people who can invent new ways of doing things, or even people who can understand and appreciate a new way of doing things." They also tend to look on Ada and new Ada-oriented software development methods with suspicion, and this can be devastating during both the proposal effort and the formal reviews. Because IV&V personnel gain much of their expertise during the review of past projects, the typical IV&V contractor will tend to lag behind the state-of-the-art contractor during a technology transition, like a sea anchor behind a straining ship.

In order to facilitate productive formal reviews, the manager should offer free training to the contracting agency, relevant user, and IV&V personnel, and also work to convince the contracting agency to make this training mandatory for their IV&V contractor. The manager should work closely with the contracting agency, and ensure that he and his developers are prepared. The manager should have decided on the project software development method prior to the time the proposal is written, and be aware of all relevant DoD and service specific policy and mandates (e.g., DODD 3405.2 *Use of Ada in Weapon Systems*) governing the contracting agency. This will help the manager know how the proposal will be judged and what to do if he wants to use Ada when the Request for Proposal (RFP) does not mandate it. The

manager should also know all relevant software development standards (e.g., DOD-STD-2167A *Defense System Software Development*), and how they should be tailored for Ada. He should work with the contracting agency to get the standards tailored to best support the fulfillment of project requirements, and because different Ada-oriented software development methods require different tailoring, no one standard tailoring is suitable for all Ada projects. When necessary, the manager should justify requests for tailoring or deviations with the Acquisition Streamlining Directive (DODD 5000.43) which prohibits the DoD from imposing "how-to-manage" and "how-to-design" requirements on the contractor. The manager can also use recommendations of industry associations (e.g., CODSIA SDS Task Force) and professional societies (e.g., SIGAda Software Development Standards and Ada Working Group). In order to educate the contracting agency and ensure that IV&V personnel do not successfully raise improper complaints against your Ada-oriented development method during formal reviews, boilerplate should be avoided and the Software Development Plan and Software Standards and Procedures Manual should be complete and specific. The manager must ensure that the contracting agency understands the benefits of the proposed innovations and the need to modify their expectations regarding project phasing, milestones, etc.

## Developers

When dealing with developers, the manager should be a leader (to new technical frontiers) and not merely an administrator. The manager needs to realize that resistance to Ada-specific ways of developing software is often due to ignorance and is a good clue to the lack of proper training, experience, and attitude of some developers. To combat this, both managers and developers must be adequately trained (both scope and depth). Failure of training results in the failure to properly exploit many of Ada's most important capabilities and in using certain language features inappropriately because of their similarities to features in older languages. This, in turn, results in the loss of many of the benefits possible with Ada. Managers should be aware that the two most common causes of trainee failure are (1) long experience with a single older language (especially assembly or C) coupled with an inflexible attitude and (2) negative management attitudes towards modern software engineering principles (e.g., "ivory tower garbage that keeps my developers from coding"), Ada (e.g., "just another oversold language being shoved down our throats"), and Ada-oriented software development methods (e.g., "bureaucratic

waste of time"). Objective tests (e.g., the Ada Proficiency Test by Psychometrics, Inc.) are available to test competency, both after training courses and prior to hiring.

Even if the manager and developers already have some Ada experience, it is often cost-effective to seek expert advice and use the experience of others to develop the project-specific Ada-oriented methods, standards, and procedures. These methods, standards, and procedures should be publicly imposed so that the developers will clearly know what is expected of them. The manager should make it obvious to the developers that he or she supports and will enforce the project methods, standards, and procedures. Once a technical disagreement has been raised, discussed, and settled, the manager should enforce it and make it clear to the developers that the methods, standards, and procedures are subject to CONTROLLED change, but may NOT be ignored. Managers should ensure that their subordinate technical managers are TECHNICALLY competent and will enforce their technical direction. Managers should keep track of their developers and use Quality Assurance as their watchdog to identify the software Luddites who never make the transition to Ada, the lone-wolf hackers, and developers with the programming-in-the-small, edit-compile-debug mentality who will undermine management's new direction. Quality Assurance and lead engineers should also be used to find any developers using Ada inappropriately or using obsolete methods and backfilling documentation to "comply" with Ada-oriented project standards and procedures. They should also watch out for new (inexperienced) developers who have been taught modern Ada methods and are then being confused or intimidated by more experienced, but conservative developers. Each project will probably have at least one reactionary individual who will always resist the Ada approaches and who must be removed or isolated for the good of the project. Where possible, such individuals should be maneuvered into working for the competition.

Managers must promote objective and egoless reporting because with Ada, they will have an even greater need for timely visibility into how the project is actually implementing the new Ada approach. Managers can ensure the necessary communication among developers and between organizations by (1) requesting staff input, (2) holding regular meetings to let developers be heard, (3) setting up a project newsletter or electronic bulletin board to spread success stories and keep developers informed and aware of the big picture, and (4) ensure that lessons learned are captured, published, and distributed to those who need them. Finally, managers must build team spirit and encourage cooperation.

## Conclusion

Remember what Ben Franklin said, "To get the bad customs of a country changed and the new ones, though better, introduced, it is necessary to first remove the prejudices of the people, enlighten their ignorance, and convince them that their interests will be promoted by the proposed changes; and this is not the work of a day." Although difficult, this is possible and even necessary. Because the entire industry is going through the same transition, there are opportunities for the innovative manager to compete successfully and move himself and his company forward. As Emerson said, "This time, like all times, is a very good one if we but know what to do with it."