The Ada Recommendation – Was It Heard?

Rayford B. Vaughn Jr.
Mississippi State University

In January 1997, the National Research Council published a report to the Department of Defense (DoD) entitled Ada and Beyond, Software Policies for the Department of Defense. *The author was a member of the committee that produced the report. This article looks at the original recommendations of the report, the process used to produce it, and comments on both intended and unintended results.*

Early in 1996, the National Research Council's Computer Science and Telecommunications Board (CSTB), began to search for committee members to serve on a committee chartered to perform a "Review of the Past and Present Contexts for using Ada within the Department of Defense." Twelve people were selected: Barry Boehm (committee chairman), Theodore Baker, Wesley Embry, Joseph Fox, Paul Hilfinger, Maretta Holden, J. Eliot, B. Moss, Walker Royce, William L. Scherlis, S. Tucker Taft, Anthony Wasserman, and me. Paul Semenza, National Research Council (NRC), was assigned to guide the committee, provide administrative support, and to serve as interface between the committee and DoD. This article outlines the deliberations of the committee, the final recommendations, and comments on what has happened since publication of the final report, entitled *Ada and Beyond: Software Policies for the Department of Defense* [1]. It represents my impressions and thoughts and is not an official opinion of the NRC, the committee members, or DoD.

The committee met in a three-day group session four times during April 1996 to October 1996: twice in Washington, D.C., once in Denver, and once in Los Angeles. Significant work was accomplished between meetings, and the committee constantly communicated electronically. During the first session, the committee agreed that the issue was far larger than just a "language decision" and needed to be taken in the larger context of DoD software engineering. It was also determined that the committee needed to hear from the software development community outside DoD. The committee discovered, through DoD speakers, that support for the Ada Joint Program Office (AJPO) was being dropped—a major concern that resulted in a finding that was considered critical to the future of the language.

Findings and Recommendations

The majority of information in this section was taken from the final report [1]. I provide editorial comments after each rationale to add to background, better understanding of the issues, or some of the influences that were present.

Finding 1: Ada Competitive Advantage

- **Finding**: Ada provides DoD with a competitive advantage for war-fighting software applications, including weapons control, electronic warfare, performance-critical surveillance, and battle management.
- **Recommendation**: Continue vigorous promotion of Ada in war-fighting application areas.
- **Rationale**: Available project data and analyses of programming language features indicate that compared with other programming languages, Ada provides DoD with higher-quality war-fighting software at a lower life-cycle cost. DoD can create a further competitive advantage by strengthening its Ada-based production factors (involving software tools, technology, and personnel) for war-fighting software.

It was understood that there was no clear definition of "war-fighting" software. Clearly, some systems are, e.g., embedded weapons systems, such as cruise missile or AEGIS guidance, and others are arguable, e.g., personnel and logistics, in a support role. Essentially, there can never be a concise definition of war-fighting software. If ample commercial-off-the-shelf (COTS) products exist that can provide the needed functionality, the system under development is probably not "war fighting" in the context of this report. Likewise, if the system requires software quality and reliability attributes higher than supportable by commercial products, the system may be categorized as war fighting.

Finding 2: Applicability of Policy to DoD Domains

- **Finding**: DoD's current requirement for use of Ada is overly broad in its application to all DoD-maintained software.
- **Recommendation**: Focus the Ada requirement on war-fighting applications, particularly critical, real-time applications in which Ada has demonstrated success. For commercially dominated applications, such as office and management support, routine operations support, asset monitoring, logistics, and medicine, the option of using Ada should be analyzed but should not be assumed to be preferable.
- **Rationale**: For war-fighting software, supporting Ada-based production factors (involving software tools, technology, and personnel) gives DoD a competitive advantage. In this domain, eliminating the use of Ada would both compromise this advantage and diminish the capabilities for maintaining DoD's existing 50 million lines of Ada. In commercially dominated areas, pushing applications toward Ada would create a competitive disadvantage for DoD.

Early in the deliberations, the committee discussed extensively the total

elimination of the mandate, particularly if DoD was intent on dropping all support for the AJPO. There were many instances of overapplication of the mandate within DoD where COTS products were bypassed in favor of more expensive Ada development for no reason other than the mandate. Subsystems were rarely considered in the context of the Ada mandate. Given that DoD would continue its support for the AJPO and that there is a stronger Ada production base here in the United States, the committee felt that an advantage accrued if war-fighting software continued to be developed in Ada.

Finding 3: Scope of Policy

- **Finding**: DoD's current requirement for the use of Ada overemphasizes programming language considerations.
- **Recommendation**: Broaden the current policy to integrate choice of programming language with other key software engineering concerns, such as software requirements, architecture, process, and quality factors.
- **Rationale**: The current policy isolates the Ada requirement and waiver process from other software engineering processes, causing programs to make premature or nonoptimal decisions. DoD has already taken steps to broaden the policy focus in its draft revision of its programming language policy (DoD Directive 3405.1).

The committee was given a draft DoD Directive 3405.1 that moved closer to a software engineering focus vs. a language-only focus. This draft was then modified by the committee and provides an appendix to the final report.

Finding 4: Policy Implementation

- **Finding**: DoD's current Ada requirement and the related waiver process have been weakly implemented. Many programs have simply ignored the waiver process. Other programs make programming-language decisions at the system level, but often a mix of Ada and non-Ada subsystems is more appropriate.
- **Recommendation**: Integrate the Ada decision process with an overall Soft-

ware Engineering Plan Review (SEPR) process. To pass such a review should be a requirement to enter the system acquisition Milestone I and II reviews covered by DoD Instruction 5000.2. It should also be required for systems not covered in DoD Instruction 5000.2 and recommended by DoD for DoD-directed software development and maintenance of all kinds.

- **Rationale**: The SEPR concept is based on the highly successful commercial architecture review board practice. The SEPR process involves peer-reviewing not only the software and system development plans but also the software and system architecture (building plan) and its ability to satisfy mission requirements, operational concepts, conformance with architectural frameworks, and budget and schedule constraints; the process also involves reviewing other key decisions such as choice of programming language.

A key concern here was the "ability" of DoD to put individuals with good software engineering backgrounds on the review boards. The review can be a powerful tool and can enforce architectural frameworks developed by the services or DoD if staffed with the right people. They can also be the "common sense" sounding board that a program manager needs when trying to make good technical and cost-effective decisions.

Finding 5: Investment in Ada

- **Finding**: In order for Ada to remain the strongest programming language for war-fighting software, DoD must provide technology and infrastructure support.
- **Recommendation**: Invest in a significant level of support for Ada or drop the Ada requirement. The strategy developed by the committee recommends an investment level of approximately $15 million per year.
- **Rationale**: With investment, DoD can create a significant Ada-based complex of production factors (involving software tools, technology, and personnel) for war-fighting application domains. Without such

support, Ada will become a second-tier niche language such as Jovial or CMS-2.

There was strong concern voiced by all committee members when it was discovered that DoD planned to drop support for Ada. Essentially, it was felt that Ada was not strong enough to stand on its own today and that the support was fundamental to its success. Great improvement had been made over the years in Ada and its support environment, and this investment would be placed at risk without continued DoD support.

Finding 6: Software Metrics Data

- **Finding**: DoD's incomplete and incommensurable base of software metrics data weakens its ability to make effective software policy, management, and technical decisions.
- **Recommendation**: Establish a sustained commitment to collect and analyze consistent software metrics data.
- **Rationale**: The five sets of findings and recommendations above are based on a mix of incomplete and incommensurable data, anecdotal evidence, and expert judgment. For this study, the patterns of consistency in these sources of evidence provide reasonable support for the results but not as much as could be provided by quantitative analysis based on solid data. A few organizations within DoD have benefited significantly from efforts to provide a sound basis for software metrics; a DoD-wide data collection effort would magnify the net benefits.

The committee found it extremely hard to find data to support any of the testimony to which we were exposed. In fact, it seems to be a systemic problem within DoD that metrics are not heavily supported and collected for review at service level or DoD level.

Reaction and Response

Although the majority of conversation and interest in the committee's recommendations has centered on whether Ada should be mandated for all system development, the report clearly goes beyond

the question of Ada and proposes that the programming language should not be the focus of concern, but software engineering practices should be. Additionally, DoD may have, over the past year, undermined the intent and recommendation of the committee in multiple areas that could put the language at risk as well as the significant investment that has been made over many years by DoD and the commercial community.

The competitive advantage that Ada gave the war-fighting community (Finding 1) was explained within the report in great detail. A key component of maintaining this advantage was continued support from a policy and financial standpoint. DoD has chosen not to invest in Ada through continued support of the AJPO (Finding 5) and not to support Finding 2, which mandated Ada for war-fighting software. The lack of support for these two essential findings thus results in an indirect lack of support for Finding 1 and its associated recommendation.

DoD's failure to support Finding 2— to mandate Ada for war-fighting software but not for commercially dominated software domains—has created concern within the Ada community; in some cases, the NRC report is incorrectly cited as the catalyst for this decision, though it clearly did not recommend such an action for reasons cited in Chapter 3 of [1].

Finding 3 appears to have been partially accepted (in principal), and DoD seems to be moving toward a process that adopts more focus on software engineering decisions vs. language decisions. Little consideration, however, is being given to changes in the system review process and the adoption of the recommended software plan review. This particular finding was important; its adoption would bring DoD more in line with current accepted commercial practice.

Finding 4 is essentially moot with the dropping of the Ada mandate. Its associated recommendation, however, contained a description of a commercial architecture review board process that was deemed necessary by committee

members. It appears that this recommendation will not be adopted by DoD.

Finding 5 was critical to the future of Ada in DoD and required strong financial backing and support for the AJPO. This finding and its associated recommendation met with strong internal opposition within DoD and in particular, disagreement between the Defense Information Systems Agency (DISA) and the Office of the Assistant Secretary of Defense for Command, Control, Communications, and Intelligence (OASD C3I). OASD C3I directed that DISA comply with the recommendation but did not seek any additional funding for DISA to do so. DISA, in a letter to the committee, outlined its position that "the AJPO was created to do a job; it has succeeded, and is no longer necessary." The disagreement between these two agencies was never resolved, the AJPO funding was not substantially increased, the AJPO director position was not filled, and the office's ability to support the Ada program became severely impaired. This course of events was consistent with the stated position of DISA in which the committee was told, "We have not determined a final date for the closing of the AJPO. We selected a date in late third quarter of fiscal 1997 as a target for planning purposes."

Finding 6 continues to be an issue today but has not been acted on in a way that makes the situation then any different today. Metrics gathering and reporting is still a problem within DoD and needs to be addressed.

Summary

It should be clear that a year after the report was released, most of the recommendations and findings have not been followed. There are earlier reports that indicate DoD adopted all but one recommendation, e.g., [3], but as can be seen from the above, little was adopted. The process changes recommended by the report seem to be under careful study by the Office of the Secretary of Defense, but the overall intention of the report was not accepted. The recommendations were meant to work together as a holistic approach to improve the software development process in DoD. A piecemeal adoption may do more harm than good. As a war-fighting language and a national competitive advantage, Ada would have to be considered in jeopardy at the current time. ◆

About the Author

Rayford B. Vaughn Jr. spent 26 years in the U.S. Army as a software engineer. His key assignments included commander of the Army's Information Systems Software Center headquartered at Fort Belvoir, Va. and the first director of the Pentagon Single Agency Manager for Information Technology Services. Upon retirement as a colonel in May 1995, he joined Electronic Data Systems (EDS) Military Systems where he served as vice president of DISA Integration Services. While with EDS, he was responsible for all EDS contracts issued by DISA. In October 1997, he accepted a position as associate professor of computer science at Mississippi State University.

Department of Computer Science
P.O. Box 9637
Mississippi State, MS 39762
Voice: 601-325-7450
Fax: 601-325-8997
E-mail: vaughn@cs.msstate.edu
Internet: http://www.cs.msstate.edu

References

1. Computer Science and Telecommunications Board, National Research Council, *Ada and Beyond, Software Policies for the Department of Defense*, National Academy Press, Washington, D.C., 1997.
2. Vaughn, Rayford, "The National Research Council's Ada Mandate Study: Insights and Recommendations," *Proceedings of the DoD Software Technology Conference*, 1997.
3. Hamilton, J.A., "Why Programming Languages Matter," *CROSSTALK*, Software Technology Support Center, Hill Air Force Base, Utah, Vol. 10, No. 12, December 1997.