# F-22 Software Risk Reduction

by Beverly L. Moody
*F-22 Avionics Software Block Lead*

*In the early phases of the Advanced Tactical Fighter (ATF) program, the Software Program Office (SPO)/Contractor Teams realized that serious efforts would need to be applied to mitigate the risk associated with developing well over a million source lines of code (SLOC) across many development sites throughout the United States, Canada, and Europe. The ATF software development team was composed of the three prime contractors, four major suppliers, and more than a dozen smaller suppliers, all of whom had to have a common goal of developing an operational flight program (OFP) for the ATF. Immediately after the contract was awarded in 1991, the Computer Resource Working Group (CRWG) and the Systems/Software Engineering Environment (S/SEE) Integrated Product Team (IPT) set out to build the framework in which the F-22 OFP would be built.*

It was almost 20 years ago when the Air Force identified the need to replace the aging F-15C air superiority fighter. In 1986, two teams received the ATF Demonstration/Validation (Dem/Val) contract: Northrop Grumman and the eventual winning team composed of Lockheed, Boeing, and General Dynamics. During the Dem/Val phase, areas of risk were identified and plans to mitigate those risks were implemented. Not surprisingly, software development risk mitigation was near the top of the list, since the ATF was to be the most software-intensive airplane built for the Air Force. The ATF Team planned to develop approximately 1.5 million source lines of code, across more than 20 software development companies located throughout the U.S., Canada, and Europe. The main focus for software risk reduction during Dem/Val was the need for a common development environment and team networks to enable rapid reliable communication across the various sites. This article focuses on software risk reduction practices in three risk areas associated with F-22 software development:

1. Development schedule.
2. Complexity of the development team.
3. Leading-edge technology challenges.

## F-22 Development Schedule

The Engineering and Manufacturing Development (EMD) contract was awarded to the Lockheed/Boeing/General Dynamics Team in August 1991. The F-22 Raptor's first flight was six years later in September 1997. Initial flight of the first avionics airplane is scheduled for late this spring, and Raptors will enter the operational environment in 2005. The time span from identified need to operational capability is almost 25 years.

The technological advances in the areas of computer hardware, software, and software development processes have been, and are expected to be, phenomenal. Keeping pace with these rapid advances over a protracted development schedule has been a continuing challenge. Major changes in the acquisition process, particularly the move from the DoD-mandated military standards to reliance on commercial standards and industry best practices, generally facilitated, but at times exacerbated, risk mitigation as contractor and government define their new roles.

### Software Development Environment

Based on experience and lessons learned from earlier programs, the Lockheed-led team wrote specifications for the tools that it thought needed to be included in the software development environment. Digital Equipment Corporation (DEC) was selected as the provider of the host environment that was VAX/VMS based. DEC was tasked to identify a set of tools that would meet the F-22 team's specifications, and to present the entire environment as a turnkey package to the team for approval.

Requirements for an interface definition and control tool were included in the specification. These were the only requirements provided to DEC for which DEC could not find a commercially available tool, so DEC committed itself to developing the Interface Definition Tool (IDT).

DEC selected Interleaf as the document publication tool, Teamwork for requirements analysis, Requirements and Traceability Manager (RTM) for requirements traceability, and Product Configuration Management System (PCMS) for configuration control of software, documents, and software development files. The common systems/software engineering environment (S/SEE) for all F-22 software development was identified and installed at multiple contractor/government sites by early 1992.[1]

In 1991, VAX/VMS was the only environment which met the team's technical requirements as well as the F-22 program security requirements for wide area networked computer systems.

Even back in the early 1990s, when we were not operating at Internet time, the team recognized that VAX/VMS days were numbered, and that the F-22 would outlive this mainstay computer system. For long-term risk reduction, the F-22 Team worked with DEC to identify a migration path that would sustain the S/SEE capability. This migration path led to a yet-to-be built environment called Cohesion that was to be based on the Alpha workstation with open VMS or UNIX. DEC never managed to bring this together in an affordable and functional package for the F-22 Team. DEC no longer exists, and Compaq, which bought them out, no longer takes orders for, or offers long-term support for, VAX computer systems. The F-22 team continues studying S/SEE migration for the near future, and will have to continuously plan to mitigate the risk of changing environments over the F-22 life cycle.

The common S/SEE and the associated networks have been invaluable to F-22 software development. The networks link the contractors, System Program Office, Air Combat Command, Air Force Operational Test and Evaluation Center, and Edwards AFB Combined Test Facility at the appropriate security levels. This connectivity provides the SPO and other government offices with nonintrusive insight as we can access common areas and desktops for documents, briefings, metrics, and software schedules.

There will always be a need for some commonality among the software teams for populating and disseminating information from the air vehicle-wide databases such as IDT and RTM. The IDT database is an integral part of building the operational flight program as it supports the trusted computing base on the Raptor. Commonality at the software code development level may not be as critical today as it was 10 years ago due to the intercommunication between computer systems, which is a lot easier than it used to be.

### Out-of-Production Processors

The old way of doing business—freeze the processor, tools, and compiler at some point, and stick with it, is no longer practical in today's high-paced developments. Processors are becoming obsolete at a rapid pace, and frequently new compilers and tool sets are not developed for old systems. Because compilers and tool sets for the new processors, which are replacing out-of-production processors, are not available in the VAX/VMS environment, the F-22 team is already using other host environments for some code development. Military developments will have to migrate processors and environments at the commercial pace or lose some of the benefits of commercial off-the-shelf products.

### Commercial Tools and Security Issues

One other lesson learned from dealing with commercial tools is that commercial tools do not inherently deal well with classified information. For example, rolling up paragraph markings to the correct markings at the top and bottom of each page is not easily done by most commercial tools. Adding a new paragraph to a baselined document may take some major editing. Coordinating/ contracting with tool developers to provide some program specific features dealing with classified markings may be necessary for future programs as well as the F-22.

### MIL-STD-2167A Mandate

At the time of F-22 contract award, software development fell under the Ada mandate and MIL-STD-2167A. MIL-STD-2167A was a standard to provide a structured process for developing software, testing the software, and documenting same. Under DoD Acquisition Reform policy, most standards were deleted and the Ada mandate was eliminated. The F-22 program deleted most of the contract data requirements list in 1996, however, most of the F-22 team software developers continue to produce the MIL-STD-2167A documentation and have it available on the S/SEE. The commercial standards that replace MIL-STD-498 and MIL-STD-2167A are based on the military standards for software development, since those standards were the best available.

### Ada Mandate

Ada 83 will continue to be the primary language used on the F-22 (80-85 percent) for the foreseeable future since much of the code is already complete. Some teams are looking at migrating their Ada 83 code to Ada 95 since most of the new compilers are based on Ada 95. To date, needed source code changes have been trivial and the migrations fairly straightforward. The promise of the portability of Ada has been validated by several programs.

## Complexity of the Software Development Team

The completed F-22 Air Vehicle OFP will contain more than 2.2 million SLOCs. There has been some growth as well as added functionality, which have added to the original estimate of 1.5 million SLOCs. Our software developers include the three primes, several major suppliers, as well as smaller single function-type developers. Some companies are developing less than 5000 SLOCs; one major supplier is developing more than 500,000 SLOCs.

### Teammates and Competitors

Some of the F-22 teammates compete against each other for other program contracts. In a highly integrated weapons system development, regular communication and coordination between designers and developers is essential. This was a new and unusual environment for most team members. The F-22 team has overcome these obstacles, which is a tribute to the various corporate management teams that have allowed their software developers to work directly with competitors in a major cooperative effort.

Ways of doing business within a company, the corporate culture, are frequently very different, and compromise has become a way of life for the F-22 team. For example, each company had its own change notice forms. Some companies used interface revision notices; others used interface change requests. Terminology and forms for a highly integrated development required F-22 specific terms and forms to aid in communication.

### Building the F-22 Team

Some of the risk-reduction efforts undertaken to bring this diverse and geographically separated team together to perform a major highly integrated software development are listed here. Lockheed /Boeing/General Dynamics selected the Software Productivity Consortium's Ada-based Design Approach for Real-Time Systems as the team software design methodology. The S/SEE Help Desk located at Lockheed in Marietta, Ga. had to support multiple time zones from Rochester, England to the U.S. West Coast. As the teams started to use the S/SEE with all the associated development tools, there were many cries for help.

Working groups were established with representatives from the primes and major suppliers to determine the processes and procedures for using the various tools to ensure a common approach was used across all the developing sites. Training in the use of the S/SEE tools was provided at various locations throughout the team.

### The F-22 Team

One outcome of all these efforts is that the team comes together (via meetings, telecons, video teleconferences, etc.) speaking a common language. Some team members share Ada package specs for interface definitions. Developers of applications that run in the Common Integrated Processor (CIP) come together in a weekly telecon with the CIP Help Desk to work through issues, problems with CIP tools such as debuggers, share workarounds, compiler experiences, and to ask for advice from other users to resolve problems or concerns.[2]

### Team-Wide Problem Reporting System

Delivering a product with a known problem at the developing site but not passing the data along to the receiving site, can lead to a lot of wasted time as the problem is rediscovered. With many developing sites, accurate and complete problem report tracking is essential. The Team's Common Problem Reporting System (CPRS) was implemented to ensure complete tracking of all air vehicle problems. Problem reports on delivered products are maintained in a single master database at Lockheed. Problems are categorized by severity, and move through various states until the initiator verifies them as closed. CPR Boards meet via telecon or VTC on a regular basis to disposition problems, determine when the fix is required, and assign the fix to a block and a particular build. Developers, labs, manufacturing, and the test pilots have access to the database, the opportunity to participate in the Boards, and have an input to the dispositioning of problem reports. Problem report closure is one of the critical metrics used to determine the readiness of a new build.

## F-22 Leading Edge Technology Challenges

### Integrated Avionics

The most challenging technical requirement for the F-22 avionics software is to successfully achieve sensor fusion. Sensor fusion combines multiple sensor target attribute data into one target track file for presentation to the pilot. Shortcomings of any one sensor will be overcome by fusion of all sensor data. Fusion has not been done to this scale on any other tactical military aircraft platform. The mission software team uses simulations and an Algorithm Prototyping and Analysis Tool to reduce this development risk.

Lessons learned from other programs told the F-22 team that it was quite common to use up the memory and throughput of the mission computer with the first delivered OFP. The CIP architecture was designed and sized for growth in functionality. Two CIPs are included in the current design with growth for a third for a minimum 300 percent long-term growth with technology insertion. There are 66 slots available in each CIP. CIP 1 has 19 slots open and available, and CIP 2 has 22 slots open and available. If necessary, but not in the current plan, more processors could be added during the EMD timeframe if the reserve required in each processor gets used up. Memory and throughput are monitored at the subsystem level on a regular basis.

### Incremental Integration

Raptor 01 OFP integration occurred in integration labs at Lockheed, Fort Worth. OFP integration included flight controls, utilities and subsystems, displays, inertial reference system, and stores management system. This OFP included more than 700K SLOCs, is flying at Edwards AFB on Raptors 01 and 02, and has nearly 600 hours of flight test. Raptor 03 had its first flight in March 2000.

Integration risk is mitigated by incremental levels of integration. Subsystem (radar, electronic warfare, communication, navigation and identification, etc.) hardware and software integration is initially accomplished at the subsystem level by the developing Integrated Product Team (IPT).[3] Software is sent to an intermediate lab or to the Avionics Integration Laboratory (AIL) at Boeing.[1] Partial blocks of software are sent to the Flying Test Bed (FTB), a modified Boeing 757 with a wing and sensor edges in addition to some of the avionics hardware, for further testing in an airborne environment.[4] The AIL has responsibility for the integration and certification of the Air Vehicle OFP for Raptor 04 through Raptor 09, as well as the production representative air vehicles and Lot 1 (Raptors 10 -27).

## Summary

Metrics is one method used to measure success in our risk mitigation efforts. Some of our earliest metrics tracked the number of calls to the S/SEE Help Desk by tool and severity (from user inexperience to immediate change needed to the tool). SLOCs were tracked, along with notes explaining any significant jumps. Subsystem metrics were kept on code and unit test, computer software component integration, and formal qualification testing. Problem report metrics by subsystem as well as metrics on AIL tasks, functions, and system acceptance test procedure completions are tracked on a weekly basis. Many metrics have changed as we moved through various phases of the software development cycle to meet the current need.

Team communication paths, whether on telecons, video teleconferencing, the S/SEE, or the PC, have played a major role in F-22 software development. Problem areas are identified early and given needed attention to reach resolution.

The Software Engineering Institute (SEI) was hired by Maj. Gen. Claude Bolton, Air Force Program Executive Officer for Fighters and Bombers, to perform an Independent Technical Assessment (ITA).[5] Their findings were very positive in the area of software risk reduction efforts undertaken by the F-22 Team. Out-of-production parts and diminishing manufacturing sources were identified as one of our biggest risks.

The next few years will be exciting and challenging times for the F-22 program. We will finish building the EMD planes, continue with flight test, move into Initial Operational Test and Evaluation, start production, and prepare to go operational in 2005. The F-22 Team expects to find that software risk mitigation efforts pay off.

## Notes

1. See page 6 for Thomas Brandt's comments on the S/SEE and page 9 for Ron Dubbs' comments.
2. See page 9 for comments on the CIP by Ron Dubbs' of Wright-Patterson.
3. See pages 4 and 5 for Maj. Gen. Calude M. Bolton's comments on the IPT as a best practice See page 8 for comments by Maj. Gen. (Ret.) Thomas Brandt. See page 9 for Dubbs' comments.
4. See page 4 for more on using the flying test bed in the F-22 program.
5. See page 9 for more on the Independent Technical Assessment.

## About the Author

**Beverly L. Moody** is the Avionics Software Block Lead for the F-22 Avionics Integrated Product Team, F-22 System Program Office, Aeronautical Systems Center, AFMC, Wright-Patterson Air Force Base. Her responsibilities include managing incremental software block deliveries to the Avionics EMD aircraft, Raptor 04 through 09. Moody began her Air Force career in 1982 at the Language Control Facility working with JOVIAL and Ada standards and compiler validations. She worked for SEAFAC performing MIL-STD-1750A computer validations and Ada training. She was then assigned to the tri-service Mark XV System Program Office as the computer resource lead working on a planned replacement to the Mark XII Identification Friend from Foe system prior to her current assignment to the F-22 SPO in 1991.

ASC/YFAAT
2130 Fifth Street
Wright-Patterson AFB, Ohio 45433-6503
Voice: 937-255-7503, Ext. 2457
Fax: 937-255-1144
E-mail: Beverly.Moody@ASC-YF.WPAFB.AF.MIL