Up Close with Maj. Gen. Claude Bolton Jr.

Maj. Gen. Claude M. Bolton Jr. is the program executive officer for fighter and bomber programs for the Office of the Assistant Secretary of the Air Force for Acquisition, Washington, D.C. His responsibilities include acquisition activities on the F-22, F-15, F-16, F-117, B-1, and B-2 programs and the Common Missile Warning System and Joint Helmet Mounted Cueing System programs. He was commissioned through the University of Nebraska's Air Force Reserve Officer Training Corps in 1969. He is a command pilot with more than 2,700 flying hours in more than 30 different aircraft, and flew combat missions in Vietnam.

CROSSTALK: Having seen the F-22 program from its beginning, what actions in the software area could have been taken to improve the current product or prevent problems you have encountered?

Gen. Bolton: We could probably use another metric or two. We had some that we got rid of and we could use a few more of them. If we had to do it over again, we would look at all of our metrics and leave them in there a bit longer until we understood what they were telling us.

CROSSTALK: For F-22-integrated avionics and software, how are you doing and how do you know?

Gen. Bolton: Really well. We have a Monthly Executive Review where we invite the key players from the contractor side and the government side. The purpose is to look at the various issues; one is how we are doing on software. We have a schedule, a block approach that we are taking.

How are we doing on the burn down? What are we accomplishing in the ground-based laboratories? How are we doing in the blocks, coding, verifying, and checking and putting that through the laboratories and so forth? That is how we track it at this level.

At the [Air Force] System Program Office, contractor and the Integrated Device Technology [level], it is a day-to-day thing; they are really into the details. We do not get into that. We do not have the time and we would just muck it up. Right now the biggest thing is the schedule. With everything we have delivered—whether to the Avionics Integration Lab [in Seattle], the flying test bed, to the test birds—over the last nine, 12 months it has been on or ahead of time.

CROSSTALK: You attribute that to these monthly and daily meetings?

Gen. Bolton: That is part of it. Most of it is because of the structure of the avionics program, the block approach[1] that we take, and that we have ground-based labs. We have one in Fort Worth; the big integration unit is in Seattle where all the pieces come together. Once it is all proven there, we put it onto the flying test bed, which is a converted 757.[2] It is all there; we duplicated the cockpit for the F-22. If you look at the flying test bed, you will find the cunard [a sensor part] up front. We have upwards of 30 engineers who look at the data streams and say 'that is what we expected, this is what we are getting, how does that compare to what we have on the ground?'

Once it is run there, we put it into the actual airplane. That process is really what is causing this success. When we look at the schedule, it allows us to ask 'how is it going, what do you need from us?'

CROSSTALK: It sounds as if you had to do it all over again, the block approach and the intensive meetings are both things you would keep. Is there anything that you would do differently?

Gen. Bolton: We put together a team [around December 1998] that emphasized the integration of the team approach. We were looking at R-19. I said, does that mean we have 19 [schedule revisions] since we started research and development? Well, yes . . . every six months we look at it and put another revision on [the schedule].

> "We made the 'i' a capital 'I' in the Integrated Product Team . . . That is paying big dividends."

We needed to be more proactive, rather than reactive and changing the schedule. Let's figure out where we ought to be. Let's put a team together across the enterprise and put the best and the brightest folks on it. We made the 'i' a capital 'I' in the Integrated Product Team (IPT). It brought more folks together. That is paying big dividends.[3] We went to R-20 and held schedule longer than we ever have [April 2, 1999 to present].

We initially were going to fly the Block 3 software in April 2001. The team looked at that. Because of all the things we were doing, because of the confidence we have in the processes we have, because the successes we have had over the last year or so, we believe we can move it from April 17, 2001 to [putting software in the aircraft] around Oct. 30, 2000. It goes back to the team understanding what is going on, and everybody working on the problem—the best way to get this sent to the aircraft and do it safely, professionally, and correctly from an engineering standpoint.

CROSSTALK: What do you believe are the key metrics for F-22 software development?

Gen. Bolton: The schedule and the cost. If it were not for the cost, this would be easy. Inside the [Washington] Beltway, for this program cost is the primary factor. Everyone is concerned about that. The contributors to cost in this program are avionics and airframe. I am very sensitive to how much it is costing, and that sensitivity is shared by the team. From a software standpoint that may not be the answer a lot of folks want, but unfortunately we cannot do the things that need to be done if we do not have the money to do them.

CROSSTALK: Did your system have any unforeseen setbacks?

Gen. Bolton: Early on in Block Ø we had some surprises. That is because we were going from a first-ever design, even though the design tools and practices environment were the best I have ever seen. You put it into some type of ground-based laboratory,

then into an integration laboratory. You say, 'let's go fly this,' and find a flying test bed.

That showed us early on that we had some problems. I would not call them major. It is a whole lot better to find it there, in the ground integration laboratory or the flying test bed. We have found most of our errors on the ground in the integration test laboratory. To give you an example: We put the radar—which is electronic, very complicated, software-intensive—into the flying test bed, turned it on, and it received targets the first go, [which is] unprecedented for radar.

"Our first best practice is the IPT. The next is the use of ground-based laboratories with the flying test bed. I am not sure how we can do this program without the flying test bed."

CROSSTALK: Did your team develop or collect best practices?

Gen. Bolton: Our first best practice is the IPT. The next is the use of ground-based laboratories with the flying test bed. I am not sure how we can do this program without the flying test bed. You have the opportunity to take from the ground and put into a flying environment all the boxes—you have the cockpit, all your engineers there, and folks looking at the data stream. That is beautiful. You can see it all right there. On a flight that can last hours you can get a hell of a lot done, far more than you can on an actual fighter and at a far reduced cost. Those are really key: the IPT, the ground-based laboratory, and the flying test bed.

CROSSTALK: What is your vision for process improvement involving the F-22?

Gen. Bolton: We go back to the team. We are always hammering the cost for this. The team was already ahead of us. There is continuous process improvement.

This process turns out not to be dependent solely on personalities. We have had Dr. Hans Mark [Director of Defense Research and Engineering] take a look at us several times because of his vast experience on other systems, particularly flight controls, avionics, software, and structures. The last time (November 1999) we spent a couple of hours with him and went through excruciating detail.

His bottom line was, 'You are doing the right things.'

Notes

1. See page 10 for comments by Ron Dubbs, F-22 Weapon System Chief Engineer at Wright-Patterson AFB on the usefulness of the block approach to the F-22 program.
2. See page 14 for more on using the flying test bed in the F-22 program.
3. See page 8 for comments by Maj. Gen. (Ret.) Thomas Brandt on the value of the IPTs. See page 9 for Dubbs' comments on the effectiveness of IPTs.

Maj. Gen. Claude Bolton Jr.
AFPEO/FB, 1060 Air Force Pentagon
Washington, DC 20330-1060
Voice: 703-588-7300

Asking the Right Questions

The Air Force had a program, now defunct, called Bold Stroke that was a two-day course for colonels and flight officers.

"The bottom line was to teach them how to ask the right questions," said Maj. Gen. Claude Bolton Jr.

Asking the right questions goes all the way to the top. Bolton cited the example of some software trouble reports he had studied that showed that a particular program was 50 percent complete. Several weeks later, the report showed the same thing: 50 percent complete.

"Wasn't that what it was two weeks ago?" he recalled asking a staff member. 'Yes sir,' the staff member replied with a big smile.

Something was wrong, the general concluded. He told the staffer he wanted the following points addressed regarding the report:

- What does the software trouble report tell me?
- How do I know?
- Show me.

"OK, you have a week. See you," the general told the staff member.

Monday morning came.

The new report showed that the program was 52 percent complete. Now they were getting somewhere, the general remembered thinking. Confident that the staffer knew the answer, the general asked what it meant for that program to be 52 percent complete.

"There was no answer," Bolton recalled. "There were a lot of words that came out, but there was no answer. It took us a month to understand what that meant. And trying to answer that question, we looked at our process in that program and with the contractor and found out that our quality control folks were not in the process. The library where we were supposed to be booking this was out of sequence. And this had been going on long before I got there. We had to get the right expertise in there. Just by going through those questions we were able to turn that program around."

Dwindling software expertise in the military is another factor in managing a software-intensive program.

"When I first got into this part of the business and started what is now the F-22, we knew we were going to have the heart and soul of the avionics," Bolton said.

"The pilot would be a manager of the weapon system. We were going to have a lot of lines of code. An expert tells me he can give me 30 lines of code and cannot test it exhaustively because it takes him a lifetime to do it.

"To that end I do not think we have come very far in the last 17, 18 years. We have put more manpower on writing the lines of code. We have somewhat better tools. But if instead of 1.9 million lines of code, I want a billion lines of code, how do we do that? Can't. And if you say that, I guarantee you that you will never have a Star Trek. You will never have the Enterprise." Bolton said.

"I am of the notion that there is a way of doing that. Software has got to get easier in terms of our ability to create it, test it, and use it.

"Military and industrial software expertise is dwindling on both sides. Our expertise has gone by the wayside and it is not going to improve. It is going to the commercial sector.

"Let us go back and look at what is causing concerns as we develop software. Is it the writing of code? Is it the testing of the code? Is it the integration? Is there a way we can engineer or manage our way out of this so we do not have to have all that expertise?

"We have not stepped up to that."

The next step in software management, Bolton concluded, is answering the question of how to continue putting out a quality product, but with far fewer resources.