Click to Print

Close

By Graham Warwick, Guy Norris
Washington, Fort Worth

Former NASA Administrator Michael Griffin believes high-profile program failures in aerospace and defense could be avoided by engineering elegant system designs. But he cannot quite define "elegance in design."

Paul Eremenko, a program manager with the U.S. Defense Advanced Research Projects Agency (Darpa), believes problems could be avoided by measuring and minimizing the complexity of system designs. But he cannot yet measure "complexity."

They both believe the systems engineering processes that have served the aerospace and defense community since pre-Apollo days are no longer adequate for the large and complex systems industry is now developing. But they have different ideas on how to solve the problem.

Systems engineering is under scrutiny after major problems leading to program delays, cost overruns and even cancellations. The issues typically lie with unintended and unanticipated interactions between elements that are only uncovered during integration, testing or, worse still, in service. And they are only getting more serious as systems become more interactive.

"How is it that we continue to encounter failure of important and complex systems where everything thought to be necessary in the way of process control was done, and yet despite these efforts the system failed?" Griffin asks. "The answer cannot lie in continuing to do more of the same thing while expecting a different outcome."

Systems engineering has been a hallmark of the U.S. and European aerospace industries since pre-Apollo space programs. "We in the U.S. and in Western democracies have demonstrated we do this better than anyone else in the world," says Griffin. "It is the ability to deliver large, complex systems and do things that have never been done before . . . [but] we have persistent and unusual failures in these large, complex systems."

Systems engineering involves decomposing a design into separable elements, characterizing the intended relationships between them, and verifying the system is built and operates as intended. It is often represented as a "V," with decomposition and design on the downstroke and integration and verification on the upstroke.

Failures usually occur at the interfaces between system elements, many times between elements thought to be separate. "There are two exponential trends, in software and network communications, that mean a lot of elements of a system can now impact each other. The result is incredible complexity: millions of lines of software and hundreds of nodes," says Jeff Wilcox, Lockheed Martin vice president for engineering.

"Historically, systems engineering has been successful in bringing order to the development of systems as they have become increasingly complicated. But there is a big difference between complicated and complex," he says. "Complicated is decomposable, which is what systems engineering is based on. Complex systems are no longer strictly decomposable, and systems engineering has to adapt."

Griffin, now professor of mechanical and aerospace engineering at the University of Alabama in Huntsville, says the failures in large programs indicate "something is missing" from the systems engineering process. "There are cases where everything thought necessary was done, and somehow something went wrong anyway."

Adding processes is not the right answer, he believes. What is needed is a new view that the core systems engineering function "is not primarily concerned with characterizing the interactions between elements and verifying that they are as intended." What's more important, he says, is understanding the dynamic behavior of those interactions.

Lockheed Martin started a systems and software engineering strategic research effort in 2007 "because we recognized complexity makes it necessary to take a new look at systems engineering," says Wilcox. As systems get much of their functionality from software, it is no longer separate and distinct. “Complexity is about the whole ecosystem and systems engineering has to become more holistic,” he says.

That ecosystem must include the global suppler base. “Because of the extended interfaces, supply-chain management becomes part of the process, and is a challenge because of the varying degrees of supplier sophistication,” he says.

Model-based systems engineering, already used in the software industry, is gaining ground. Compared to documents, models can be more effective conveyors of information throughout the supply chain. “Models give contextual information. With a model you can understand the degrees of freedom and the interactions,” he says.

“We build sophisticated and accurate models now, but we need to evolve to where these can be passed from the government to the primes and on to second- and third-tier suppliers,” Wilcox says. “We have to add the capability for those suppliers to access that kind of environment.”

The new approach proposed by Griffin hinges on identifying and quantifying the key attributes of what he describes as “elegant” system design. Selecting elegant designs and rejecting those that are not will be not easy, he admits “Elegance in engineering design is an ineluctable concept. It is immediately apparent when it exists, yet it is difficult to define, impossible to quantify and, so far, apparently incapable of being taught.”

What are the characteristics of an elegant design? “First, it has to actively work and produce a result that’s what you intended,” Griffin says. Second, the design must be robust. “The space shuttle operated as intended will produce the most amazing results, but get outside the envelope and bad things happen. That’s not a robust design.” At the moment there is no way to measure robustness, he says.

Third, it should be efficient. “But how do we evaluate this?” asks Griffin, who adds that the community’s ability to predict how well a design will meet objectives is “poor and, at best, pathetic.” The ability to differentiate between and among competing designs, based on cost, schedule and performance predictions, is even poorer, he says. “We need better means of comparing designs before developing.”

Finally, Griffin says systems engineering should “minimize unintended consequences. We rarely ask ‘What is this going to do that I don’t want it to do?’ We don’t model or predict it, and I’ve never been on a design committee where that question was asked,” he says. “It is a rare modeling and simulation tool that provides any analysis of potential side effects in addition to the primary purposes for which it is developed.”

This is where Darpa comes in, with an effort to reshape the industry that is bold in its vision and sweeping in its scope. The agency’s Adaptive Vehicle Make program portfolio encompasses an entirely new approach to conceiving, designing, developing, verifying and manufacturing aerospace and defense systems. The overall goal is reducing the time its takes to deliver a complex system like a vehicle or aircraft by a factor of 5-10.

Strongly influenced by the integrated-circuit industry’s success in managing complexity, the portfolio includes a new systems engineering approach that would separate development from manufacturing, do away with the industry’s traditional design/build/test/redesign cycle, enable verification without fabricating prototypes and ensure the first unit off the production line works as intended.

The enabling element of the portfolio is the Meta program, which aims to enable “fabless” development of systems that are probabilistically verifiable as correct by design, compressing the cycle time by removing manufacturing from the loop. And key to Meta is the development of a metalanguage—like the metadata that make digital files easier to archive and retrieve—that will characterize components and their interactions.

“We will achieve that by paying attention to the interactions, by publishing and subscribing to a contract on how a component will affect its neighbors,” says Darpa’s Eremenko. “The metalanguage will properly characterize everything up front, and we will have a library of tens of thousands of components and how they behave that we can build up into a system,” he says.

To build a system, engineers will check component models out of the library, each with metrics that allow overall system flexibility and adaptability to tracked as the design is composed. “The metalanguage enables potential system interactions to be identified up front, so we can design a less complex system,” says David Corman, chief engineer for network systems technology for Boeing Research & Technology. “A new design flow and tools keep track of complexity and allow us to build a system that is correct by design, with software and hardware behavior that is verifiable to a certain probability level.”

Component characteristics are now represented in domain-specific tools. “The metalanguage will capture all of them, and integrate the individual representations to enable anyone to do analyses across interactions” says Doug Stewart, Boeing’s principal investigator for Meta. “As they build a component, they will build a model in the metalanguage and put it into the component repository. If we can bring cycle times down and create new systems rapidly, we can make component reuse more attractive.”

Under Meta, Boeing leads one of the teams developing “concrete, scalable and composable” metrics for complexity and adaptability. “Engineers look at complexity, and do not purposely design systems to be complex,” says Corman. “But we have to establish a formal, mathematically precise mechanism to measure complexity and adaptability . . . [where] adaptability means the system elements have sufficient margin, and can serve multiple purposes.

“The types of systems we need to meet the most demanding requirements tend to be more complex and harder to develop. The interactions between different system elements are not necessarily well understood until a significant way through design and testing,” Corman says. “Meta looks at how to account for flexibility and adaptability in system definition, and how to assess for complexity and design for adaptability.”

Wilcox says, “Design for simplicity is a very important concept. Engineers like to take a set of performance specs and design from a clean sheet. But there are a lot of ways to solve a given problem, not all with the same degree of complexity, and some have been solved before perfectly adequately. We need to make sure engineers have access to a library of proven patterns and reference architectures as building blocks,” he says.

“Reuse also is a powerful concept,” says Wilcox. “Kelly Johnson and Skunk Works used it to develop the U-2 in nine months and the F-117 in 18. They pushed the state of the art where they had to meet one important performance specification and the rest could be an 80% solution. Industry needs to make sure it takes advantage of this approach.”

A second part of Darpa’s drive to revolutionize manufacturing is the Instant Foundry, Adaptive Through Bits (iFAB) program. “With Meta enabling fabless design, iFAB is what a factory will look like that can build anything within a wide product domain, any ground or air vehicle if you give it the Meta characteristics of the design,” says Eremenko.

“We are not creating any new piece of manufacturing equipment, it’s about the information flow that ties it together,” he says. “We usually wrap factories around products, like the Joint Strike Fighter—iFAB is a programmable factory that can build anything.” Additive manufacturing can build almost anything from a 3D model with zero learning curve, he says, though there will still be manual labor in final assembly.

“We need to break the paradigm of long cycles from design to product,” says Dave Neyland, director of Darpa’s Tactical Technology Office. There is skepticism within industry about the agency’s approach, and he sees that as a challenge the program must overcome to succeed. “With constraints on defense budgets and national economies, everybody is on the alert, looking for a different approach,” he says.

“Imagine the industry doing ‘make before buy,’ which every industry does except defense,” says Eremenko.

Photo Credit: Lockheed Martin

Photo: Lockheed Martin