

System Engineering: What it is and What it is Not

Michael D. Griffin

**King-McDonald Eminent Scholar
University of Alabama in Huntsville**

**Technical University of Catalonia
15 December 2009**

Most of you will have heard of Baron Charles Percy (C. P.) Snow, and will know of his observations on the breakdown in communication between the humanities and the sciences. Trained as a scientist, Snow served as Minister for Technology under Prime Minister Harold Wilson, yet was more famous as an author, with sixteen novels and eight works of non-fiction to his credit. He would be near the top of nearly any list of scientifically literate authors, or of literarily-talented scientists. Snow developed his theme in *The Two Cultures and the Scientific Revolution*, in 1959, and explored it further in *The Two Cultures and a Second Look*, in 1963. He decried the decline in standards of higher education, and in particular what he viewed as the almost willful ignorance by the modern cultural elite of scientific fundamentals. In a summary of his theme, Snow noted,

“A good many times I have been present at gatherings of people who, by the standards of the traditional culture, are thought highly educated and who have with considerable gusto been expressing their incredulity at the illiteracy of scientists. Once or twice I have been provoked and have asked the company how many of them could describe the, Second Law of Thermodynamics, the law of entropy. The response was cold: it was also negative. Yet I was asking something which is about the scientific equivalent of: 'Have you read a work of Shakespeare's?'

I now believe that if I had asked an even simpler question — such as, What do you mean by mass, or acceleration, which is the scientific equivalent of saying, 'Can you read?' — not more than one in ten of the highly educated would have felt that I was speaking the same language. So the great edifice of modern physics goes up, and the majority of the cleverest people in the western world have about as much insight into it as their Neolithic ancestors would have had.”

While Snow's criticisms did not go unanswered – most famously by literary critic F.R. Leavis – the essential truth of his observations was, and is, widely acknowledged. His elucidation of the “two cultures” has become a societal paradigm, a bumper-sticker phrase to describe the basic cultural separation between the arts and the sciences that is clearly visible to most of us. Even those who know nothing else of Snow's work are probably familiar with this one phrase.

Today, I want to discuss the two cultures that, if we think about it, we find embedded in the profession we call ‘engineering’, and how we are linking them, and must link them, through the discipline known as ‘system engineering’, a product of the American aerospace sector.

Let us first explore the nature of the “two cultures” in engineering. I have always loved the view of the engineering profession captured by the great Theodore von Karman when he said, “Scientists study the world as it is; engineers create the world that has never been.” Less eloquently, engineers are designers; they synthesize knowledge to produce new artifacts. Von Karman speaks to what most of us, and certainly most laymen, would consider the essence of engineering: engineers create things to solve problems.

But all of us who are engineers know that the engineering profession also has a rich scientific side, the analysis of these artifacts and the prediction of their behavior under various environmental and operational conditions. Adapting von Karman's observations, it may be said that *engineering science* is the study of that part of the world which has been created by man.

Sadly, many students have been led to believe that engineering science *is* engineering! In a curriculum of 120 or more credits leading to a bachelor's degree in a branch of engineering, the typical student is required to take one, or maybe two, courses in design. Everything else, aside from general-education requirements, focuses on the analysis, rather than the creation, of engineered objects. Graduate education often has no design orientation at all. So, engineering as taught really deals with only a part of engineering as it is practiced.

This trait is so pronounced that engineers who have spent their careers – even widely-recognized careers – in design and development, focusing on the creation of objects rather than the creation of papers for publication in refereed journals, are essentially unemployable, hence unemployed, in academia. No matter how well credentialed a practicing engineer may be, when the inevitable

search committee meets to rank the applicants for a department chair, or a tenured position, it is a rare designer who can offer even the minimum of “academic” qualifications expected of an applicant for the position of assistant professor.

Some universities have recognized this inherent bias and its consequences for the training of their students, and have sought to remedy it by creating titles such as “Professor of Practice”, or similar appellations. But it is a truism that the longer the title, the less important the job. So this term serves only to emphasize the point that these particular faculty members are not “real” professors, hired and promoted on their merits in a straight-up competition among all candidates. One wonders if this is the message we really want to send to those who will design – or not – the world of the next generation.

But if the present excessive focus on engineering science in the engineering curriculum is of concern, it is nonetheless true that the fundamental difference between modern engineering and that practiced prior to the Enlightenment is the development of formal analytical methods and their application to man-made objects. This has allowed the prediction of performance, and the limits of that performance, in the environment in which a given device must function. It has allowed the refinement of designs through methods more sophisticated than the trial-and-error techniques to which our ancestors were limited. It has enormously shortened the time required for a design cycle for the objects we create. A control system engineer might say that the formal methods of engineering science have produced an enormously improved feedback path for the engineering design loop. More simply, engineering science has taken engineering beyond artisanship.

But, interestingly, the development of formal methods has not altered in any way the fundamental nature of design, which still depends, as it did in antiquity, upon the generation of a concept for a process, technique, or device by which a given problem might be solved. The engineering sciences have provided better, and certainly quicker, insight for the designer into the suitability of the concept than can be provided solely by building it and examining its performance in its intended application. But a human being must still intuit the concept. We have no idea how we do that. And until we do, we have little hope of developing a formal method by which it can be accomplished.

It must be said that some progress in this area has been through research into “genetic algorithms”, which use the tools of engineering science and mathematical simulation to explore the consequences of iterative random changes to a given design. The performance of the design is evaluated against objective criteria. If a

change results in a net improvement it is retained; otherwise, it is discarded. In this manner, the design “evolves” to a higher state of suitability to its intended “environment” through the pressures of artificial, rather than natural, “selection”. Modern engineering analysis tools offer the ability to conduct what is essentially a very large number of randomized design cycles in an acceptable period of time.

But this process does not seem, at least to me, to be much akin to the intuitive synthesis of a human brain when it leaps almost instantly from a perception of a problem to an idea for its solution. “Creativity”, used in this sense, remains thus far the sole province of biological computers.

So at least for now, there remains an artistic side of engineering, and it is fully as much an art for its practitioners as any painting, sculpture, poem, song, dance, movie, play, culinary masterpiece, or literary work. The difference between the cultural and engineering arts lies not so much in the manner of creation of a given work, but in the standards by which that work is judged. In the humanistic disciplines, human aesthetics sets the standard by which merit is assigned to a finished product. In the end, aesthetic sensibilities vary with place and time, and are ultimately matters of opinion. The role of opinion in evaluating a work of engineering is, by comparison, much restricted. In engineering, more objective methods are employed to judge the degree to which the completed work meets the standards established for it, or fails to do so.

This brings us to the role of failure in engineering design. Regardless of the sophistication of the analytical methods brought to bear, they are applied to a theoretical model of a device operating in a theoretical model of the real world. The model is not reality, and the differences produce opportunities for the real device to fail to operate as intended in the real environment. An evolutionary biologist might say that the gap between model and reality is an environmental niche in which failure, like a new species, can thrive.

Civil engineer and author Henry Petroski has, in a series of essays and books, explicitly noted the crucial role of failure in producing ultimately successful designs. In *Success Through Failure: The Paradox of Design*, and other works, Petroski establishes the point that new designs, or successive iterations and refinements of a basic design, have as their essential purpose the elimination of failure modes known to be inherent in earlier designs. He further argues, by means of many examples, that designers must go beyond merely ensuring success; they must strive to anticipate the ways in which a design might fail. Great designers

and successful designs incorporate, in advance, methods to mitigate such anticipated failures.

But in recent decades human artifacts have become increasingly complex, building upon and extending former art and, especially, combining disparate elements of established art in new ways. This has been accomplished at an astonishing pace, a cause and a result of Moore's Law, the approximate two-year doubling time of computational throughput, which has held sway for several decades. While a large bridge cannot properly be considered a "simple" structure, involving as it does the interaction of thousands of component parts, it clearly pales in complexity relative to, say, a space shuttle, which relies for its success upon the interaction of millions of parts derived from a dozen technical disciplines.

Failure in complex systems can arise in so many more ways than in simpler systems that the quantitative difference ultimately produces qualitatively different behavior. It becomes unreasonable to expect, other than through the harshest of hindsight, that a particular failure mode might have been or ought to have been anticipated. Indeed, results from the modern study of complexity theory indicate that complex systems can experience highly non-linear departures from normal state-space trajectories – i.e., "failure" – without anything being "wrong".

Among the first to study complex engineering systems was Charles Perrow, in the landmark work *Normal Accidents*. Perrow argued that adding additional processes, safety measures, and alerts to complex systems – the traditional design approach to improving system safety – was inherently flawed, because for complex, tightly-coupled systems and organizations, failure is inevitable.

Perrow is a sociologist, not an engineer, but his points are well taken. Those of us who are aviators, or who are familiar with the history of aviation, can point to numerous high-profile accidents where the crew became occupied with minor anomalies and their warning systems, only to fly a perfectly good airplane into the ground. Most of us can also cite analogous incidents from other fields.

Yet, we have evolved complex systems for good reasons, and we will clearly continue to do so. The modern air transport aircraft is an incredibly complex device, and the system within which such aircraft operate is far more so. But in the last five decades this system has revolutionized world society, culture, and economics. It will not be shut down merely because it cannot be made perfectly reliable. Nor will we do so with any of the other complex appurtenances of modern society which did not exist a century ago, but which are now deemed

essential. So, if we are not to eschew the use of complex systems, how do we make them as reliable as possible?

I believe that the answer to the above question is “system engineering” which has evolved as a discipline of modern engineering from its roots in the American aerospace system development culture.

System engineering and its allied discipline of systems management are treated from a historical perspective in the excellent text by Stephen Johnson, *The Secret of Apollo*. Johnson retraces Petroski’s path, showing the development of system-oriented disciplines to be the natural reaction to the failure of early, complex aerospace systems, including large aircraft, ballistic missiles, and spacecraft.

From its first introduction into the engineering lexicon, “system engineering” has been a question-begging term. In earlier times, it was considered by many in the traditional engineering disciplines to be a category without a subject matter. Even today I find the term to be, in my opinion, misused and misunderstood by many who claim to be practitioners of the art. So, having spent what I believe to be the most productive part of my career as a system engineer, let me say a few words about what I believe system engineering is, and what it is not.

System engineering is the art and science of developing an operable system capable of meeting requirements within imposed constraints. The definition is somewhat independent of scale, and so these words are useful only if one understands that it is the big-picture view which is taken here. We are talking here about developing an airplane, a spacecraft, a power plant, a computer network. We are not talking about designing a beam to carry a particular load across a known span.

System engineering *is* a holistic, integrative discipline, wherein the contributions of structural engineers, electrical engineers, mechanism designers, power engineers, and many, many more disciplines are weighted and considered and balanced, one against another, to produce a coherent whole that is not dominated by the view from the perspective of a single discipline. System engineering *is* about tradeoffs and compromises, about generalists rather than specialists.

System engineering is *not* about the details of requirements and interfaces between and among subsystems. Such details are important, of course, in the same

way that accurate accounting is important to the Chief Financial Officer of an organization. But accurate accounting will not distinguish between a good financial plan and a bad one, nor help to make a bad one better. Accurate control of interfaces and requirements is necessary to good system engineering, but no amount of care in such matters can make a poor design concept better. System engineering is about getting the right design.

Complex systems usually come to grief, when they do, not because they fail to accomplish their nominal purpose. While exceptions certainly exist, it remains true that almost all systems which proceed past the preliminary design phase will, in fact, accomplish the tasks for which they were explicitly designed. Complex systems typically fail because of the *unintended* consequences of their design, the things they do that were not intended to be done. The Second Law of Thermodynamics is sufficient to guarantee that most of these things will be harmful! I like to think of system engineering as being fundamentally concerned with minimizing, in a complex artifact, unintended interactions between elements desired to be separate. Essentially, this addresses Perrow's concerns about tightly coupled systems. System engineering seeks to assure that elements of a complex artifact are coupled only as intended.

C.P. Snow believed that mutual comprehension and appreciation between the arts and the sciences, which had existed in earlier times, had been erased by his time. He did not find a means to restore it. I sometimes think that the gap between synthesis and analysis in engineering is as wide as that between the arts and the sciences of Snow's "two cultures". But the fact remains that designers simply do not think or work in the same way as analysts, and this does on occasion produce a certain cognitive dissonance. When it occurs in the context of a complex system development, catastrophe is a likely result.

System engineering is the link which has evolved between the art and science of engineering. The system engineer designs little or nothing of the finished product; rather, he seeks a balanced design in the face of opposing interests and interlocking constraints. The system engineer is not an analyst; rather, he focuses analytical resources upon those assessments deemed to be particularly important, from among the universe of possible analyses which might be performed, but whose completion would not necessarily best inform the final design. There is an art to knowing where to probe and what to pass by, and every system engineer knows it.

Like other branches of engineering, system engineering has evolved out of the need to obviate dramatic failures in complex systems. Such failures are not new. One of my favorite books is a fascinating text entitled “*Structures: or, Why Things Don’t Fall Down*”, by Prof. J.E. Gordon of the University of Reading, England, written in 1978, at the end of Prof. Gordon’s long career as a structural analyst. It is aimed at a level appropriate to an intelligent technical professional in any field. I recommend it highly. Regarding the matter of spectacular engineering failures, I quote Professor Gordon (pps. 352-353):

“... there are, of course, a certain number of great dramatic accidents which, for a while, monopolize the headlines. Of such a kind were ...[numerous disasters follow] ... These are very often intensely human and intensely political affairs, caused basically by ambition and pride. ... One can at once recognize a certain inevitability about the whole procedure. Under the pressure of pride and jealousy and ambition and political rivalry, attention is concentrated on the day-to-day details. The broad judgements, *the generalship of engineering*, [my emphasis] end by being impossible. The whole thing becomes unstoppable and slides to disaster before one’s eyes. ...”

In thirty-six years of engineering practice, of many kinds and in many situations, I have not seen a more appropriate assessment of what is truly important in engineering. We must of course get the details right. However, to be a complete engineer, one must also master what Professor Gordon calls “the generalship of engineering”.

I will be frank. Educators, and I include myself, are far less certain how to teach “generalship” than we are of how to teach the laws of thermodynamics. And yet it is clear that an understanding of the broad issues, the big picture, is so much more influential in determining the ultimate success or failure of an enterprise than is the mastery of any given technical detail. The understanding of the organizational and technical interactions in our systems, emphatically including the human beings who are a part of them, is the present-day frontier of both engineering education and practice.