Larger font
Printout - copied
from Email
Message

SYSTEMS ENGINEERING SEEN FROM AFAR

John N. Warfield
December 30, 1996

Richard, I have been reading your ongoing dialog on issues related to systems engineering, along with comments of others that you have kindly sent to me. You asked for comments, and I am sitting here trying to compose some. Since I have not done any systems engineering since roughly 1968, I feel well-qualified to use the above title–having been away from it in both space and time, but still feeling that I can perceive it in a galaxy far, far away.

Gödel's famous theorem. I want to start this discussion by mentioning Gödel's famous theorem to the effect that for any formal language, there will be statements creatable in that language which can never be proved in that language.

Famous Booboos. Much is made of this theorem by many. I suspect that it is covertly the basis for ignoring western logic in most of what passes today for science. As you may know, Whitehead and Russell tried to prove that all of mathematics could be constructed with only western logic as a basis, in the Principia Mathematica. Some years after that was published, a well-known Harvard professor announced that they had succeeded in doing so. Unfortunately, some would say, some years later still the theorem mentioned above hit the mathematical literature.

Will the Real Godel's Theorem please stand up? As for me, I believe that Godel's theorem is (a) a wonderful feat of mathematical accomplishment, (b) a vastly overrated result in terms of using it as an excuse for failing to give western logic its proper due in day to day matters, (c) possibly a good metaphor to use with respect to debates going on within a discipline about that discipline by members of that discipline (please allow me to pretend that systems engineering is a discipline with members, at least for a few short paragraphs).

Physician, heal thyself--or the man who is his own lawyer is a fool. In this latter respect, I am inclined to think that the worst people to make judgments about systems engineering are people who do systems engineering or people close to it in spirit.

Internal Linguistic Pollution–or Has Systems Engineering Created Its Own Everglades?
After all, it has become what it has become under the aegis of those people; and if it has flaws it probably indicates that it suffers from "linguistic pollution" produced within the commnity of systems engineers; i.e., from a language that muddies up the air and water to the extent of possibly being toxic: ergo maybe Godel's theorem, loosely applied and conceived, offers a rationale for looking at systems engineering from afar.

Electrical Engineering and Other Great Fields. Allow me to digress (please). As I look at electrical engineering (in which I claim to have been educated), I note that it has certain attributes, including these:

(1) A Basis in Science. All of the EE's who went to school with me learned where the scientific basis for EE came from. Names like Ampere, Volta, Ohm, Kirchhoff, Maxwell, etc., etc., were commonplace, and others such as Heaviside and eventually Laplace were household

names. We knew that these people had conducted major laboratory experiments, developed measuring instruments, created systems of units, etc., etc.

2) Foundation in Formalisms. The term "formalism" apparently was invented by a well-known mathematician called David Hilbert. I like it better than "mathematics". To me it means both (a) a mathematical system constructed axiomatically, with a major body of deductible propositions coming out of a modest number of carefully and thoughtfully stated assumptions; (b) any component of a larger system that does not deal with anything specific other than symbols and manipulations of them.

For example, x = y/z to me is a formalism, embedded within a larger formulism called "algebra". Likewise, the logic equation f = x1x2 + x3 is a formalism, embedded within a larger formalism called propositional logic, or Boolean algebra, or something akin to that.

3) Foundation in Empirical Evidence. We all learned in my student days about the formalisms that underpinned electrical phenomena. We sometimes learned the origins of those formalisms, and we gained much confidence in using them, because they had a property of definiteness about them that was often lacking in casual conversation in the pool hall. Moreover, we were able actually to test their utility in the laboratories where we studied empirically the same electrical phenomena that the early giants discovered for us.

4) Integrating the Foundations. We learned that Science, Formalisms, and Empirical Evidence, all coupled together, made up a formidable body of knowledge unlikely to be challenged; though always subject to improvement in precision and quality.

Later as we moved into the digital age, we saw an invention, the transistor, whose performance could only be understood by applying formalisms of both the continuum type and the statistical type; with more modern giants such as Fermi and Dirac and our own John Bardeen conceiving and fabricating ultra small devices that now form the basis for major high-tech industries, wonderful employment, etc., etc.

WHAT IS THE POINT OF THIS?

Tell me where, in systems engineering, one finds the combination of Science, Formalisms, and Empirical Evidence to underpin this area of study?

OF COURSE, I MUST CLAIM TO KNOW THE ANSWER. Here it is:

One cannot find them, or even traces of any one of them, because one has never looked, and does not know that it must be done.

NOW, HAVING LOOKED MYSELF, ONCE AGAIN I PROFESS THE SECRET.

The Shoulds. The Science that SHOULD BE the cornerstone of systems engineering is behavioral science, because our greatest problems today with systems are problems of human behavior.

The Formalisms that SHOULD BE underpinning systems engineering today are (a) the formalisms of Western logic and, most especially, the DeMorgan Theory of Relations, which I have placed in the service of humanity in an algorithmcalled "Interpretive Structural Modeling" and (b) any of the more quantitative formalisms from any field of math or science that enjoy particular relevance to some portion of that aggregation of relationships that we call a system.

The empirical results that SHOULD BE underpinning systems engineering today are (a) the measures of complexity that I have introduced (the Miller Index, the Spreadthink Index, the DeMorgan Index, and the Aristotle Index) because the provide precisely the results we need to divest ourselves of the loose cannon or underconceptualized work carried out today in the absence of the underpinning that is required to give this field substance and (b) any experimental results gathered from anywhere and adequately interpreted that give us insight into the complexity which the indexes clearly show is there.

Foucault and Peirce: The Primers of Our Existence. Michel Foucault (the late French scholar of the history of thought) and Charles Sanders Peirce ("America's greatest thinker") have collectively provided the essential philosophical basis for a rebirth of systems engineering (by whatever name). In his Archaeology of Knowledge, Foucault sets the tone by noting that we need to go back and dig up where "the knowledge" originated, and the conditions under which it reached its current state. Peirce enlightened us by focusing upon "the four methods of fixing belief"--Authority, Tenacity, Metaphysics, and Science; and by showing us clearly where to put our trust.

Armed with Foucault's attitude of inquiry, and choosing Peirce's categories, we could conceivably carry out this project:

(1) Construct the set of propositions upon which systems engineering is purported to rest.
(2) Decide, for each proposition, whether the belief that produces this came from Authority, Tenacity, Metaphysics, or Science.
(3) Tote up the statistics. How many of them are untainted by the historical dig? How many come from Authority? how many from Tenacity (we've always done it that way)? How many come from Metaphysics (a priori)(sounds good, even though there is no empirical evidence in the archives to support it)?
(4) After the description and the diagnosis, then come up with a prescription. Try to avoid the common lingo of the systems engineering community.

IN CLOSING.

Your comment about "project manager" has to be challenged a bit. As you may know, to be a "program manager" in DoD, you had to be certified by Henry Albert's Defense Systems Management College (set up initially by David Packard who was appalled at the lack of systems thinking in defense acquisition), and in being certified you came about as close to a career slot as anybody who gets a B.S. in Systems Engineering in Dean Sage's program.

How do you like them apples? Will you send it to your colleagues and let them shoot at this?

Best,

John Warfield

PS As you know, I am leaving for Ghana. Don't expect me to respond to criticism any time soon.

JNW