Derek Hitchins
Systems Scientist
17 December 2020

Two "Systems Engineering"…

There seem to be two quite different concepts.

I've been attending a two-day conference. Well, I say "attending"—remotely *à la mode,* of course. But, I must say it was pretty well done. And, under the circumstances, enjoyable. I mean—straight out of bed, in front of the desktop, morning coffee in hand? A 'lightly boiled' to follow… What's not to like…?

Anyhoo. It was the annual Complex Systems Design & Management (CSDM) Conference, ostensibly in Paris and/or Madrid, but with speakers from all over, including an excellent contribution from Pittsburgh, in spite of their impending snow storms. And Systems Design (SD), complex or otherwise, is contentious in some quarters… Correct me if I'm wrong, but INCOSE, does not *appear* to consider SD as systems engineering. Which, as it turns out, is quite interesting. Particularly for me, a one time Systems Design Manager in a pre-eminent Systems House.

As is so often the case, the two days of the conference had somewhat different flavours. The first day saw forward-looking, innovative, "big picture" work. And 'systems design' tended to start with trying to understand the customer's problem, conducting operations analysis, etc., to get a handle on a functional design concept-in-context that would solve the customer's problem. In one approach, the functions were performed by discrete assemblies, so building a physical assemblage of functional chunks. And holistic design was on display, too—designing a total energy system as a whole, to be robust, resilient and future proof. Most impressive.

The second day presented differently. So much so that it took a while to understand what was going on. The main theme appeared to be complexity, brought about by Cartesian reduction of the whole into many, many parts, and how to manage that self-induced complexity. So there was a need for a common language, to be provided by **RDS**-CW (Reference Designation System for Construction Works), or IEC/ISO 81346.

This worthy tome declares itself to be about ‘systems thinking,’ where everything is a system. So, for an aircraft, the fuselage is a system, the wing is a system, the spoiler in the wing is a system, the flap is a system, each internal door is a system, indeed every component appears to be a system. (Remind you of the fleas? “Big fleas have lesser fleas upon their backs to bite ‘em. And lesser fleas have smaller fleas, and so *ad infinitum.”)*

Looking at the examples given in the introduction to the ISO, it appears to be a well-regulated approach to physical decomposition (i.e. Cartesian reduction), designed to provide unique designations for parts. Which is good. Invaluable even.

But is it “systems?” *Prima facie,* it looks to be straightforward reduction, i.e. the opposite of *holism*—which is at the core of systems thinking, systems design and, consequently, systems engineering—

parts of a whole are in intimate interconnection, such that they cannot exist independently of the whole, or cannot be understood without reference to the whole, which is thus regarded as greater than the sum of its parts.

Can a flap, a strut or a component can be reasonably defined as a system, i.e. *“a complex, organized whole of material or immaterial things.”?* And can reduction, the antithesis of holism, be considered as systems thinking?

Complexity is a problem, but one that has been comprehensively addressed in connection with the human body, by anatomists and medical folks. And the human body appears to be one of the most complex systems around…as one conference presenter indicated, presenting its twelve (or is it thirteen?) subsystems. You remember: Nervous, Digestive (my favourite at Christmas), Circulatory, Respiratory, Immune (important for COVID-19), Excretory and so on. Notice they are all functional, not physical—and there is no decomposition. None can exist, except in the interactive presence of all the others…{Could you consider an aircraft in the same fashion? Functional subsystems, not physical parts? You would then see those functional subsystems "flowing across" the physical parts, ignoring their boundaries.}

Levels of Organization/Integration

Biology/Anatomy		*Man-made Systems*	SE Layer
Biosphere	IX	Nation	*5. Socioeconomic/societal SE*
Ecosystem	VIII	Organization	*4. Industry Systems Engineering*
Community	VII	Company	*3.Business Systems Engineering*
Population	VI	Group	
Organism	V	Platform	*2. Project Systems Engineering*
Organ System	IV	System	
Organ	III	Subsystem	*1. Product/Subsystem Engineering*
Tissue	II	Composite	*Artefact Engineering*
Cell	I	Component	

Complexity

* Population - all the organisms that belong to the same species, in the same geographical area
** Community - a group of interacting living organisms sharing a populated environment

Numbers refer to 5-layer SE Model:see Hitchins D.K. (2003) *Advanced Systems Thinking and Management,* Artech House, MA

One approach to managing complexity is to view Levels of Organization. The left hand column above shows the anatomist approach, that many of us learned at school. At the bottom, Level I, is the cell, the smallest living thing, and from which all others are made. The human cell is a highly complex system in its own right, a factory in miniature, and there are many different kinds of cell, each with its own emergent properties. A number of different cells combine their emergent properties to create a Tissue, Level II; tissues are quite distinct structures—flesh, bone, nerve, all are tissues. The emergent properties from a variety of different tissues come together to create an organ, Level III. And so on.

The third column shows a corresponding hierarchy, Levels of Integration, for manmade systems, while the fourth column indicates how we tend to view the associated systems design/engineering activities.

All of which is logical, rational, reasonable, and allows you to manage complexity. How? Well, if you know the emergent properties of a cell, say, you really don't have to know about the complexity within it. That complexity is "encapsulated." Similarly if you know the emergent properties, behaviours and capabilities of a man-made component, say a microchip, you really don't have to know about the complexity within it. It is also "encapsulated." And so on, up the two parallel hierarchies. Think about it—when you see a plane flying in the sky, how much do you have to know about the complexity within in it?

And that is how "Systems" manage complexity—by "bottom-up" synthesis, rather than "top-down" decomposition.

There were second-day presentations on agility, visions of the future, architecting, tools, product line engineering, and then I finally twigged. The second day was largely about designing methodologies for managing the complexity resulting from decomposition and Cartesian reduction. And, since any methodology may be defined as a "system of methods," the presentations could be reasonably identified as about "systems design." Not so much the design of end-user systems, as the design of systemic methods for enhancing manufacture, assembly and integration of end-user machines, artefacts, and systems.

Couldn't help remembering the apocryphal story of Apollo, where engineers had been so busy decomposing the many and various systems into parts, and parts of parts, with master reference drawings for so many things that they had reached *impasse*—they had created so much detail and work for themselves that they could not cope.

At which point, so the story goes, systems scientists were called in, used synthesis to manage the complexity, and "set the show back on the road." How? Well, isn't that what systems design is all about—holistic synthesis?

* * * * *

A good, thought provoking conference, delivered under trying circumstances. Well done everyone.