# Unraveling Systems Engineers from Systems Engineering: frameworks for describing the extent, variety and ambiguity of systems engineering and systems engineers

Hillary Sillitto
Systems Engineering Director, Thales UK
1 Linthouse Road, Glasgow G51 4BZ, Scotland, UK

**Abstract:** This paper addresses a number of long-running debates about the scope and boundaries of systems engineering, and about competency, specialisation and differentiation of "systems engineers" within these boundaries. Many authors have proposed segmentation of systems engineering based on one or multiple axes of differentiation. Some emphasise the need for "T-shaped people", expert in one area of engineering but able to work across many. This paper builds on the existing literature to set out an integrated framework for understanding the internal "flavours" and specialisations of systems engineering, within the overall scope and boundary of systems engineering as defined by the BKCASE SEBOK. The paper is intended to inform senior leaders in systems engineering organisations responsible for organisational and workforce development.

## INTRODUCTION

Five major drivers are forcing the systems engineering community to provide a clearer definition of the extent and limitations of what systems engineering is and can deliver, and what systems engineers do. These drivers are:

- Increasing complexity of real world problems and coupling between systems of all sorts – including technical, organisational, societal and environmental systems –creates a new demand for people who can apply a "systems approach" to provide coherent and effective solutions to complex large-scale multi-domain problems

- The need to improve organisational performance leads to a focus on lean thinking for end-to-end value (including supply chain integration and through-life service provision), on specialisation to improve efficiency, and on "T-shaped people" (Elliott et al) who can clarify and align purpose and provide leadership across diverse stakeholder groups

- The need to extend current systems engineering frameworks to include sustainability considerations means that systems engineers need to expand their system boundary to include "eco-systems" and "geo-systems".

- Systems and Software engineering "coming together": interdependence of SE/SW for a very important and increasingly widespread class of complex system solutions, and a wide dissatisfaction with the effectiveness of the interface between the two disciplines, create a demand for improved skills in software definition among a large subset of the SE community

- The overlap between systems engineering and programme/project/engineering management leads to lack of clarity on the unique technical content and added value of some key systems engineering roles, which we find are often performed poorly or not at all.

Can we expect Systems Engineers to master both societal and environmental complexity and the intricate skills required properly to define the system requirements for software? This would imply a very wide scope of generalisation; whereas the need for better organisational performance might be best satisfied by increasing specialisation. A very important community is looking to apply a systems approach and systems thinking to a wider range of complex large-scale socio-technical problems and to engineering domains that do not involve software. Yet the "burning platform" creating an urgent demand for change in many SE organisations is increasing dissatisfaction with the interface between systems engineering and software engineering, brought into sharp relief by rapid deployment of model based software engineering and agile software processes, neither of which fit well with traditional systems engineering process models that focus on documents and review processes. In many organisations it seems that the workload on systems engineers is increasing due to increasing demands to service programme and customer demands, and they therefore have less time to attend to the needs of their software colleagues and to acquire the skills necessary to manage this interface better.

This paper sets out a framework within which a sensible debate can be conducted about how to manage these issues. It is presented in three major sections:

1. Models that explain and structure the extent, variety and ambiguity of systems engineering (the "what?")
2. Models that explain the different flavours of systems engineering practice and professional attainment (the "how?")
3. A discussion of the difference between "what systems engineers do" and "the cross-disciplinary landscape that needs systems thinking and a systems approach" (the "who?").

## Models that explain and structure the extent and variety of systems engineering ("the what"?)

### *Core models*

(Ring et al, 2008) identify "extent, variety and ambiguity" (EVA) as the three key characteristics of systems engineering. (Godfrey and Blockley) insist that "ambiguity" has to be handled by recognising and accepting uncertainty and change rather than trying to legislate them away through tight contracts; there are successful precedents for complex systems being delivered on time and on budget using collaborative "win-win" commercial constructs that align incentives to focus on common purpose, where all parties are incentivised to deal with risks and problems as soon as they emerge in the way that minimises total cost to the enterprise. Ambiguity in a situation can be measured: for example Warfield's "Spreadthink index" (Warfield, 2006) measures the alignment or otherwise of different experts' views of the relative importance of different aspects of a complex situation.

Derek (Hitchins, 1995) described five types of system – subsystems and technical artefacts, project systems, business systems, industry systems and societal systems: this is a convenient measure of the "extent" of a system. The ISO 15288 standard (ISO, 2008) sets out a process framework for managing the complete lifecycle of a system - another dimension of "extent". Dave Stupples (Elliott et al, 2007) proposes that the "variety" of systems engineering can be measured in terms of "within a discipline", multiple disciplines", and "socio-technical integration".

## *Extending existing models to include sustainability and environment*

The INCOSE UK SEASON report (Sillitto et al, 2009) provides a 3-dimensional model which combines the Hitchins extent and Stupples variety axes with a notional lifecycle based on the ISO 15288 process architecture. In discussions about an appropriate model for the Sustainable Systems Module at the University of Bristol, Sillitto and Godfrey extended all three axes of the SEASON report model to account for sustainability considerations (Sillitto, 2009). The additions are as follows:

The Hitchins 5 layer system classification is extended to two further layers beyond "societal" systems: "ecosystems" – characterised as anything containing DNA; and the "geo-system"- the physical universe. The Stupples three level model is extended to a fourth: environmental integration. And the 15288 lifecycle model is extended to include re-use or recycle.

These extensions allow all factors affecting sustainable systems to be considered within recognised and mutually compatible systems frameworks. It is important to realise that while they are represented as orthogonal for simplicity, reality is more complex; and of course the need for a holistic understanding permeates all layers and dimensions of this model.

It has been suggested that Hitchins levels 1-5 are qualitatively different from the extensions because they represent purposeful manmade systems while eco and geo systems just "are". Leaving aside the obvious theological debate, ecologists do "engineer" ecosystems, and as we move to geo-engineering we will also be attempting to engineer the energy balance of the earth - indeed we have already re-engineered it without particularly meaning to. Conversely, many societal systems (and engineering systems of systems!) are emergent and are not consciously engineered.

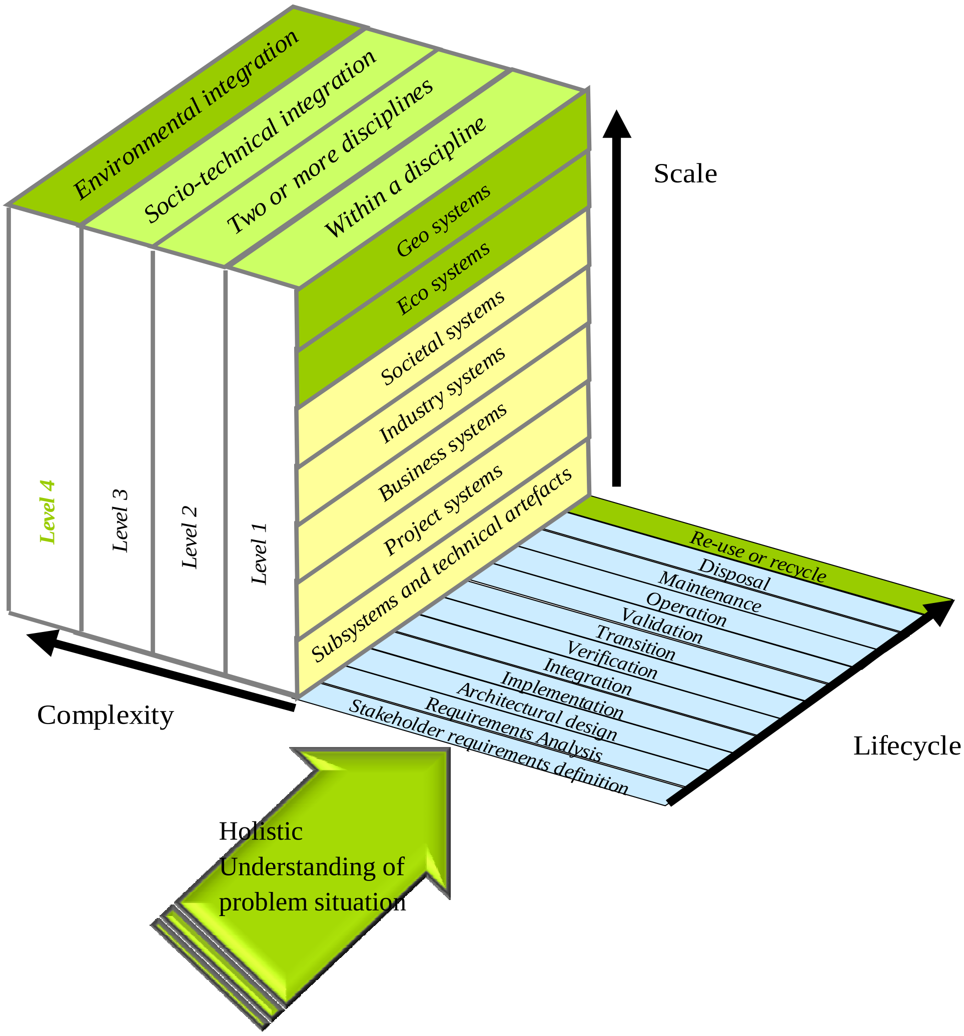

Figure 1: Sillitto/Godfrey extensions to Stupples, Hitchins and ISO 15288 models

## *Characterising variety*

We can now map out the implications of the different "Stupples levels" and start to see what this means in terms of the technical, mission-domain and integration knowledge and understanding involved in operating at the different levels. For the sake of completeness, the aperture of "systems engineering within a discipline" includes not only classical "hard" system disciplines (structures, hardware and software, etc.) but also "soft" system domains (organizations, systems of business processes, supply chains, economic systems, political systems, ecological systems, etc.).

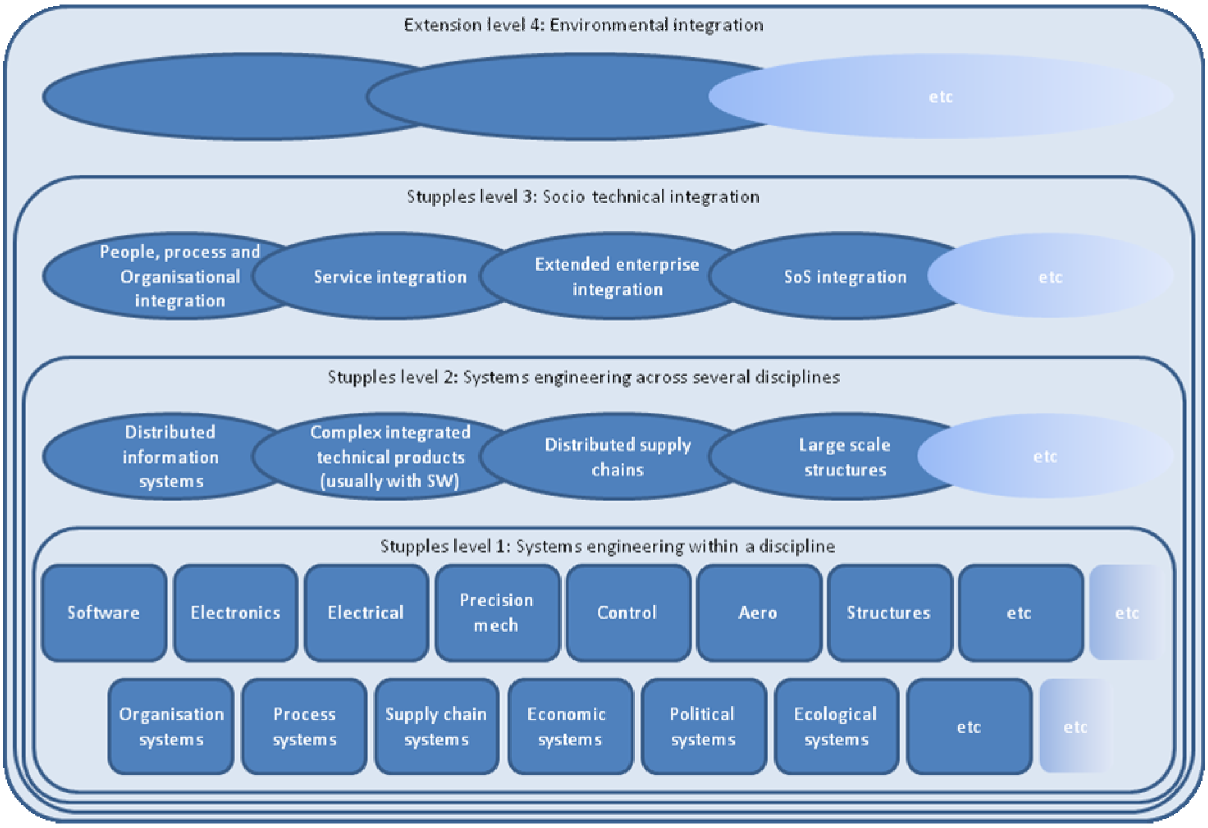

Figure 2: Extended Stupples levels

**Systems engineering within a discipline:** Stupples Level 1 is concerned with applying a systems approach and systems engineering methods within a discipline or technology. All of the individual disciplines in level 1 may operate at an "extent" that requires a systems approach and systems engineering methods. So the concept of "systems engineering within a discipline" is valid, and would be expected to involve a high level of specific domain skills. We can view the software component of a distributed information system as a distributed software system in its own right. Similarly, the "30 bolts" that hold an aero engine to the airframe (Price & Beasley) need to be designed as part of a complete mechanical system to maintain the structural integrity and performance of the whole aircraft. Many university courses with "systems engineering" in their name are concerned with Level 1 SE, applying systems approach within a discipline.

**Systems engineering across disciplines:** Stupples Level 2 involves systems engineering across several disciplines and is exemplified by:

- "product systems engineering" at Hitchins level 1;
- the classic "project systems engineering" role at Hitchins level 2.

The nature of interdisciplinary SE is different at the two Hitchins levels.

Systems Engineering at Hitchins Level 1 – referred to above as "product systems engineering" - involves satisfying user needs or a contractual requirement statement through the optimal (or "satisficing") integration of multiple technologies into a complex product or subsystem. Systems Engineering at this level requires the ability to engage in detail with the technology experts and understand the trade-offs within and between technology domains. In products with a direct interface to the user, human factors and usability should be included in the inter-disciplinary skill-set.

The Stupples Level 2 role at Hitchins level 2 is the classic “INCOSE project systems engineering” role in those cases where the human and organisational aspects of the operational system are outside the boundary of the system of interest for which the project is responsible. This role will be less focused on technology integration and more on creating the system architecture to achieve the required system level behaviour and performance from available or feasible subsystems. Again, human factors and usability should be included in the inter-disciplinary skill-set.

In systems involving software, a critical skill for “Stupples level 2” systems engineers is the ability correctly to scope, bound and specify the software content of their system and to integrate the often quite disparate lifecycles of the software element with the lifecycles of the host hardware and the overall system. There is often a tension between a view of software as a component within individual subsystems, and the software system layer though the whole system. Scalability and system of systems integration are best served by viewing software as a coherent “layer” through the whole system; whereas programmatic pressures often lead to treating the software of each subsystem separately, which may lead to integration and maintenance problems downstream in the form of higher support cost and limitations on wider system-of-systems integration.

**Socio technical systems integration:** As societal complexity increases, the dominant systems problems are increasingly seen as social and organisational rather than technological; and systems skills are seen as increasingly relevant in domains outside the traditional areas of SE application. Many technical systems that met their specifications have been unsuccessful in operation. Many expensive software features languish unused; whereas simple features in systems prove to be unexpectedly usable and popular (for example SMS messaging in mobile phones, and chat-room technology in distributed command and control systems). Introduction of new systems usually involves matching changes in culture, business process, information flows, people skills and organisation. Stupples level 3 implies a skill in operating across the socio technical boundary and being able to ensure that a new system will be successfully transitioned from technology development to operation within a business or enterprise to provide a useful service that delivers value to stakeholders and society.

There is an increasing need for socio-technical systems integration in both customer organisations (as new technology systems need to be increasingly integrated with changes in people, process and organisation to be effective) and supplier organisations (as contracts increasingly demand that suppliers take responsibility for system operation and service provision as well as or instead of system development).

Few people whose main experience is in Stupples level 1 or 2 roles will comfortably slot into level 3 ones; a different set of skills and behaviours is required to operate at Level 3. Even fewer people, however widely skilled, will be able to handle both the socio-technical and the software-intensive aspects of systems integration in a project simultaneously. Yet a holistic approach to integration and synergy over this whole scope is what is required for successful complex system development.

Socio-technical integration skills are relevant at all Hitchins levels. They are the main issue at Hitchins levels 3-5 (business, industry and societal systems); and increasingly important, as we have just seen, at level 2 (project systems). Socio-technical integration is also relevant in many Hitchins level 1 systems. Any major product innovation requires a good level of socio-technical skills to understand how users are likely to respond to and interact with the product. In many innovative product systems - the Apple product line is a particularly obvious example - a disruptive technology or user interface can change consumer behaviour and develop new business models. This also reflects an important aspect of ambiguity in systems engineering:  the requirements for new products are often difficult or impossible to establish without an iterative “exploration” between what is possible and what is useful.

**Environmental integration:** Some engineering domains are accustomed to environmental integration – civil, oil and gas, for example. Environmental integration is becoming increasingly important in three respects: environmental impact, sustainability of resource usage, and unintended consequences of coupling between apparently unconnected developments. For example engineering projects may affect water flows across wide areas, pollution incidents may propagate through air or water across wide areas, and water usage depletes aquifers that cross national borders.

We illustrate this with two vignettes showing how the Sillitto/Godfrey extensions of the Hitchins 5-level model allow us to properly account for sustainability issues and environmental impact. Both address the coupling between Hitchins level 1 subsystems and technical artefacts, and level 6/7 eco and geo systems.

**Sustainability of product development:** A product such as a mobile phone involves integration of multiple technical disciplines (software, optics, electronics, mechanical, materials science, power storage, radio, - -) to create the handset – a Hitchins level 1 system. A level 2 project system is required to put in place the network infrastructure – base station, switching, billing systems etc – to support mobile phone users. The infrastructure is funded by a level 3 business system whose viability depends on making money from providing a service to individual consumers. So the primary source of requirements for the infrastructure and its interface with the handsets are driven by the business model for the network provider. Costs to the consumer and to individual network providers are minimised by common standards used throughout the worldwide industry system (Hitchins level 4). These standards have supported an innovative and adaptable business model which supported a startlingly rapid adoption of mobile phones worldwide and allowed usage models and business models to emerge that were not anticipated by the early developers of the “system”. Mobile phones have had a major impact on societal systems (Hitchins level 5) by allowing people to communicate in ways that were previously impossible, notably when on the move, and by allowing rapid deployment of phone services in sparsely populated areas where wired telephony services would not have been economically viable. Mobile phone systems interact with the geo-system (level 7) in two important ways: the effect of physical geography on base station coverage; and the use of or hard-to-extract materials such as rare earth metals in batteries and other critical components.

**Environmental impact of failures in Hitchins level 1 systems:** The recent oil spill in the Gulf of Mexico illustrates another form of coupling through all of the levels from technical artefacts (Hitchins level 1) to geo-system (level 7). A mechanical failure in a subsystem (level 1) led to an unplanned release of crude oil into the sea (both part of the level 7 geo-system), with consequential widespread damage to wildlife and vegetation (eco-system, level 6) and major disruption to human society over a large extent of the Gulf coast (level 5 societal system).

The failed component was part of a drilling rig system (level 2 project system of which humans were a critical component, so involving Stupples level 3 socio technical integration) operated by a subcontractor (Hitchins level 3 business system) on behalf of the oil company. The overall drilling operation involved a complex web of support and service contractors so was a level 4 industry system. The blow-out destroyed the rig and killed eleven of the crew, so the immediate effect of the failure was a human tragedy in a Hitchins level 2 system. The wider impact was societal and environmental tragedy at levels 5 and 6, all due to the unplanned and undesirable coupling between the failed (level 1) technical artefact and the geo-system (level 7). During the recovery operation, there was extensive use of remotely operated underwater vehicles, and the ability to seal the well successfully depended critically on low level design characteristics (MMI, positional accuracy) in the ROVs that determined the finesse with which the problem could be diagnosed and attacked.

## *Recapitulation: Relationship between Hitchins and Stupples levels*

Superficially it may seem that there is equivalence between Hitchins levels and Stupples levels. However this is not the case. It would be more correct – though probably still an over-simplification - to suggest that they are orthogonal. As stated earlier, the Hitchins levels relate to “**extent**” of the system while the Stupples levels relate to “**variety**”. A rough correlation between them is shown in the following figure (Fig 2). This can be illustrated with a single pair of contrasting examples.

First consider an extended command and control system. Let us assume it comprises a new set of software applications deployed on an existing computer and network infrastructure. This could be a major project in its own right, so in terms of extent it is a Hitchins Level 2 “project system”. Whereas, superficially at least, it would often be considered purely as a software system, so in terms of variety it is a Stupples Level 1 system. (The need properly to consider the socio-technical systems engineering aspects of large software systems is increasingly understood but often not properly taken on board. All large software systems SHOULD be regarded as Stupples Level 3 socio-technical systems engineering problems unless there is no change to people, process, training, social, cultural, policy or organisational factors. See, for example, Sillitto 2010.)

Now consider a simple "technical artefact", a Hitchins level 1 system such as a mobile phone. My mobile phone integrates the following technologies and engineering disciplines (and probably more): optical, mechanical, electronic hardware, advanced batteries and power management, software, firmware, RF, thermal management and human factors. It also depends on a worldwide supply chain. It is clear that leading mobile phone vendors (one might cite Nokia and Apple as particularly noteworthy) pay great attention to socio technical engineering in terms of appearance, usability and wider social factors – key engineering requirements for mobile phones seem to include being "cool" and supporting open innovation business models.

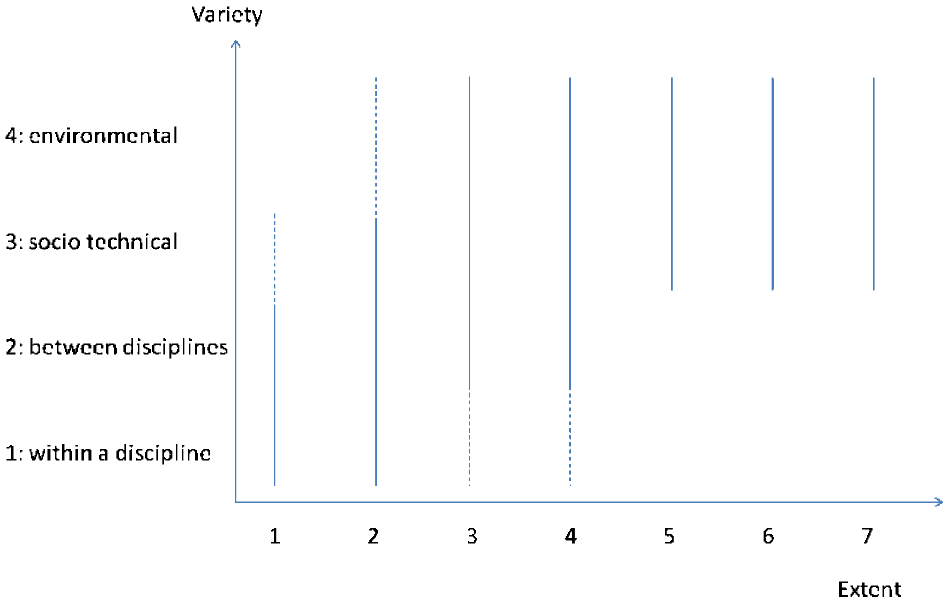

Figure 3: relationship between extended Hitchins and Stupples levels

## *Extending lifecycle models to account for sustainability*

According to ( Head), humanity has already expended 1/3 of the earth's "natural capital", mostly in the 200 years since the start of the Industrial Revolution; and at current trends will exhaust the remainder within another 200 years. This suggests that in future we will have to consider sustainability (including resource usage and waste disposal) as an element of economics, of accounting, and of systems development. So systems engineering needs to develop the tools to handle sustainability issues as part of systems engineering.

This view requires two aspects. The first is through-life resource accounting: ideally materials will be recycled and energy use will be sustainable and carbon-neutral over the full lifecycle of the system. So resource, energy and carbon accounting should be considered. The second is the ability to have coherent models that allow proper tracing of energy and resource usage through all 7 extended Hitchins levels. As an example of this, the sea has often been regarded as an infinite source of fish and an infinite sink of waste. It is now widely accepted that fish stocks are being degraded, possibly irrecoverably in some areas, both by over-fishing and by environmental pollution.

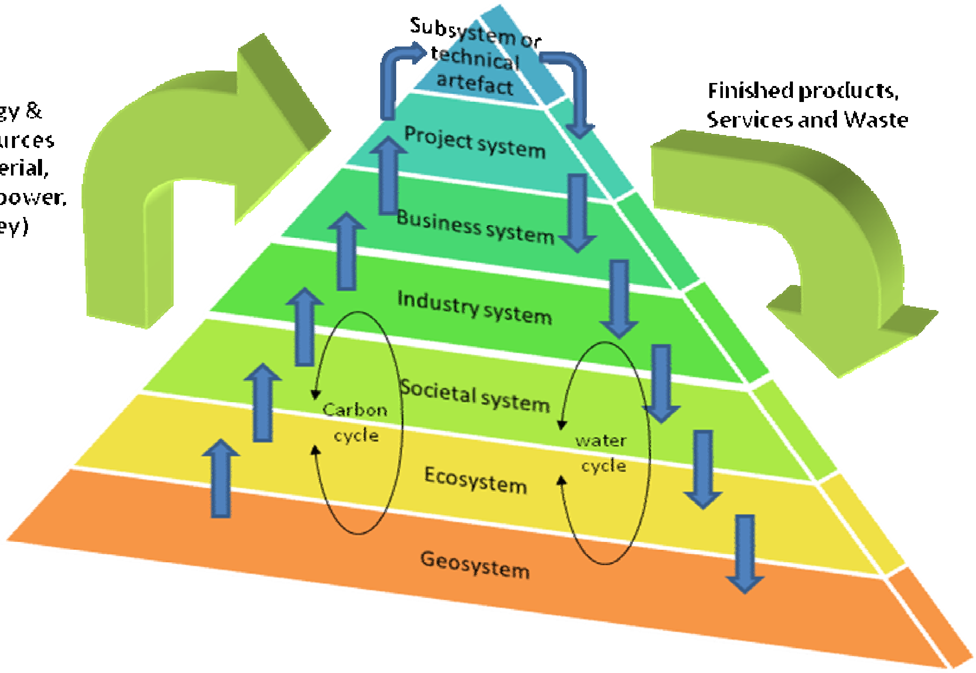

Figure 4: Tracing material and energy through the extended Hitchins levels

## Different “flavours” and levels of practice and attainment (the “how”?)

The previous section considered the domains within which systems engineering is conducted and the ways SE and systems may be categorised in terms of variety and extent. We now consider the relationship between various proposed methods of categorising systems engineering activities and professional engineering structures. This discussion considers the full scope of the “systems approach” set out in the BKCASE SEBOK (BKCASE, 2010), so considers activities required for a “systems approach” that are not necessarily performed by systems engineers. (The third and final section will elaborate this by trying to draw a clear distinction between “what systems engineers do” and “what is the wider participatory landscape”?)

This section considers four perspectives:

- a spectrum of activities ranging from “applying a global systems approach to complex problem situations”, through “inventing/architecting/designing new systems”, “implementing and managing the global systems engineering process for system development” to “practicing a specific systems engineering process”;
- looking at the wider education community, the six education attainment levels of Bloom’s Taxonomy, originating from the educational community and proposed for use to define target attainment levels in GRCSE
- a number of ways proposed to assess SE competence and attainment levels, particularly the INCOSE Competency Framework’s four levels (aware, supervised practitioner, practitioner, expert) and the five levels proposed by (Hitchins and Kasser);

- looking at the wider engineering community, the professional levels recognised by engineering institutions – the three levels used in the UK (Chartered, Incorporated and Technician) are chosen as an example.

I will suggest that many of the debates raging in the SE community can be moved forward if we consider the mapping between these four perspectives.

## *Levels of complexity and ambiguity*

Jack Ring's Systems Value Cycle (Ring 1998) shows that a systems problem first shows itself as dissatisfaction with a "community situation". This implies a high level of ambiguity and the probability that we are dealing with a complex "wicked problem". If there is an aspiration to make intelligent improvements while avoiding unintended adverse consequences, there is a need for professionals who can assess the whole problem situation and decide whether there is an opportunity to change the situation, and if so develop a suitable intervention strategy taking a holistic approach and selecting an appropriate system boundary – large enough to achieve the desired effect and manage unintended consequences, not so large that the task is infeasible. So at this stage of the lifecycle of the potential new system, we need people who can apply a global systems approach to complex problem situations. Their necessary competences are "a way of thinking", and the ability to work from first principles. (Elliott et al) refer to these people as "T-shaped".

The next stage is to invent, architect and design the new system – normally considering all aspects of the problem including social, cultural, people, organisation, policy, process and information, as well as (possibly) technology. This is the role of the System architect working initially at Stupples level 3. Once the elements of the system are established and agreed, specialists in the different domains will elaborate the solution until there is sufficient confidence in feasibility and affordability to commit to designing and building the solution.

To take the project (or portfolio of projects) through to fruition, we need practitioners who can define, implement and manage the global systems engineering process for system development using the selected technologies and organisations.

Within each systems engineering process activity, there will be specialists who have the expertise necessary to set up and run the process, do the work and produce the outputs, involving and co-ordinating the many different stakeholders who may need to participate in the process.

These four categories are shown in the next figure. Of course there is much overlap between them, and people who focus on one role in one project may focus on a different one in another. Also, the model applies recursively at multiple levels of the problem/solution hierarchy and in multiple domains of expertise.

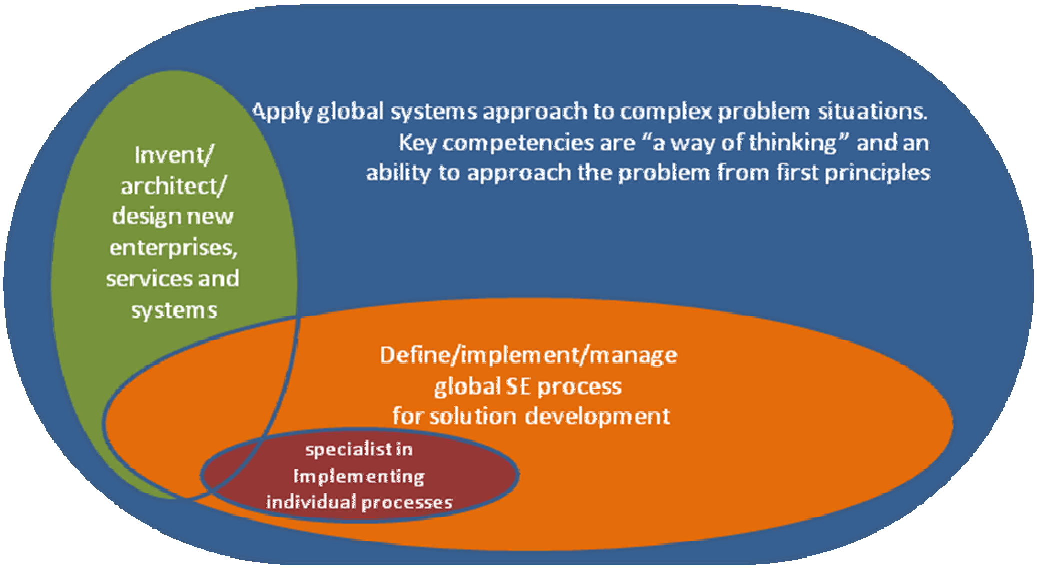

Figure 5: different roles in solving complex system problems

In a previous paper (Sillitto 2010) the current author has drawn a distinction between two aspects of the solution for complex problems. The whole problem is assumed to be a "wicked" problem, that is, one that cannot be solved but must be managed. It is assumed that we can decompose the solution into two parts, one that can be "solved" once it has been scoped and specified, and the other part that must continue to be "managed"; it is assumed that this management is facilitated by the "solution" to the "solvable" part of the problem. Relating this model to the current discussion, the critical skill in applying a "global systems approach" is to be able to make this decision about what part of the problem (if any) can and should be solved by developing a new technical solution (or process or service, or all three), and what part of it has to be "managed", and how. This requires a range of skills to be exercised at a high level in conditions of high ambiguity, the outputs of which include definitions of sub-problems that reduces the level of ambiguity (and variety and extent) faced by the other actors in the downstream part of the process. The inventor/architect role interacts strongly with the "global systems approach" role because what is feasible in terms of solution architectures strongly influences the global approach. Once the solution is designed in principle, stakeholder interests change from a demand for innovation to a demand for assured delivery. So the emphasis of the implementation team must change correspondingly, without losing sight of the overall vision and how the solution will solve the problem – not of course the original problem situation when the job started, but the problem situation faced by the stakeholders when the new solution becomes available. So a key part of the "global approach" is to understand the dynamics of change in the problem situation and build sufficient flexibility into the solution that it can adapt to the circumstances into which it is deployed, and continue to evolve to maintain its utility through life.

It is possible to plot the relationship between these four roles against the two axes of EVA (Extent, Variety and Ambiguity) and process. The diagram below is only intended to be very loosely illustrative and to pave the way for the next part of the discussion. The assumptions behind the positioning on the graph are as follows. The "global systems approach" role requires a substantial element of process to handle the stakeholder relationship issues, while also facing the highest level of EVA. This process may be largely political in nature. The main job of the architect/inventor is to invent, so this role scores high on EVA and lower on process – though Blockley (Blockley, 2010) emphasises the importance of understanding “the system as a process”, and architects need to be deeply familiar both with the operational processes that the system will support, and with the development processes the architect needs to engage with to see his or her ideas brought to fruition. The “global SE process” role bridges the gap between invention and implementation and may have to adapt known processes to the unique circumstances of the situation, so faces a considerable level of EVA. The “process specialist” (a specialist in a specific SE process area) does his normal job when the project needs it done, so is high on process and low on EVA.

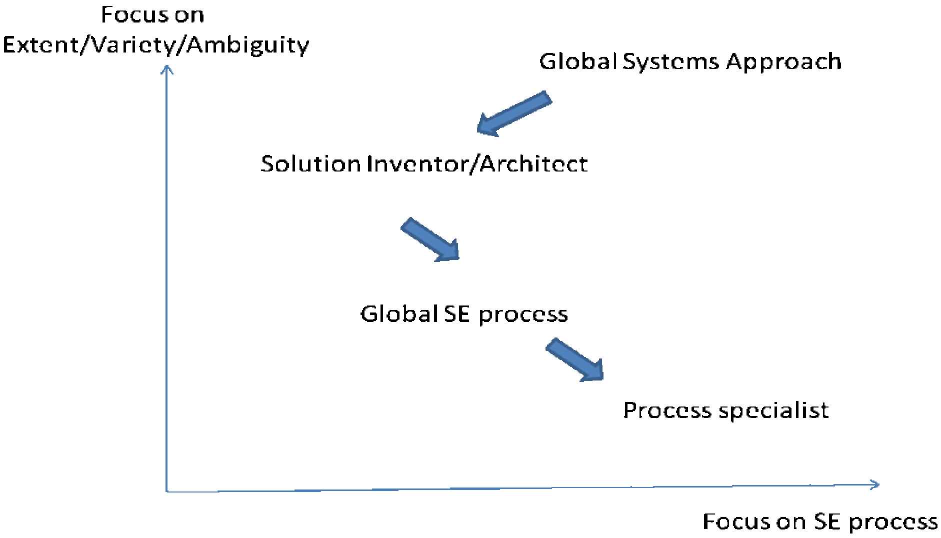

Figure 6 The four roles of the previous figure plotted against EVA and process focus

## ***Skills and Attainment levels in the Educational community***

Bloom’s Taxonomy of educational attainment levels has six levels from 1 (has knowledge) through 3 (can apply) to 6 (“Evaluation” or “Creating”), and has been proposed as a metric for student attainment target and achievement in the GRCSE element of the BKCASE project.

The original Bloom’s taxonomy for the cognitive domain (Bloom, 1956) is shown in the following table alongside Anderson and Krathowl’s proposed changes (Anderson & Krathowl, 2001): the differences are discussed and explained by (Atherton).

| Bloom, 1956 | Anderson & Krathowl, 2001 |
|---|---|
| Evaluation | Creating |
| Synthesis | Evaluating |
| Analysis | Analysing |
| Application | Applying |
| Comprehension | Understanding |
| Knowledge | Remembering |

## ***Methods proposed and used within the SE community***

The (INCOSE 2010) Competency Framework assesses a number of skill areas against four levels: “aware”, supervised practitioner, practitioner and expert. A number of organisations (Stevens, Presland) have found it desirable to add an extra level between “practitioner” and “expert”, referred to for example as “leading practitioner”. These are measured for each of a set of competency areas (INCOSE 2010).

The INCOSE certification process aims to certify an individual as a “Systems Engineering professional”. For the CSEP level, this involves an on-line test and a review of accredited professional experience. Against the Bloom taxonomy, it seems that the on-line test for CSEP measures knowledge – Bloom level 1 – and the review of experience measures the ability to apply that knowledge in a real work situation, so demands at least Bloom level 3. The ESEP criteria demand a much higher Blooms taxonomy level.

(Hitchins and Kasser, 2009) have published a five-level scale for systems engineers’ competence. Comparing the Hitchins/Kasser/Massie (HKM) scale against the Blooms levels, the highest level (5) of the HKM scale demands a high Blooms level and a high ability to deal with EVA.

Now returning to Figure 5 (above) we can see that the high-level roles – “global systems approach” and “solution inventor/architect” demand high ability to deal with EVA, require a variety of systems skills (covering both systems engineering and systems thinking) to be mastered at high Blooms Taxonomy levels, and equate to levels 4-5 on the Hitchins/Kasser scale. The more process-oriented roles exposed to lower levels of EVA can be performed effectively with lower Blooms Taxonomy levels of mastery (or with a high level of mastery of a narrower set of skill areas), and by people who would be scored at lower Hitchins/Kasser levels. This does not necessarily mean they are any less skilled overall, just that they may have a different skills mix.

The important point is that anyone working in any of the four roles shown in Figure 5 may call themselves a systems engineer, or something else, depending on the custom within the organisation(s) they have worked in. People and organisations that use a narrower definition of “systems engineer” may find that “systems engineers” are not able to do what they are expected to do; they may be either over- or under-qualified depending on their background and stakeholder expectations.

There has been much debate on this for a long time. The BKCASE project is allowing us to get a much better understanding of the issues, thanks to its success in building a wide and agreed statement of scope of systems engineering, and relate this to a graduate curriculum for systems engineers. It is not BKCASE’s job to resolve the debate itself; individual organisations and professional societies will have to make decisions based on the improved understanding BKCASE provides.

## *The "Professional Engineer/Technician Engineer" spectrum*

The previous sections refer to the systems engineering and general educational literature. It is instructive to compare the conclusions with existing wisdom and assumptions about professionalisms in a wider engineering community. Here we take one example of a professional engineering structure, the one defined by the UK Engineering Council. It is instructive to see how this structure maps to the previous discussion.

The three professional levels recognised by the UK engineering institutions (ECUK) are:

- "professional engineer", recognised by Chartered status (Chartered Engineer, or CEng) and requiring the ability to work from first principles and to innovate to solve unprecedented problems and create new knowledge;
- the "incorporated engineer", requiring an ability to apply a process or other known approach with intelligence and competence in a way appropriate to context;
- and the "engineering technician" who applies a specific process skill (which may be very well developed and valuable) in accordance with specified standards.

These distinctions are seen as very important in some domains of engineering and less so in others. They also have a strong notion of potential. CEng can be achieved at quite an early age (four years after graduation) if the applicant has clear potential to develop rapidly, as evidenced by rapid progression and high attainment in education and early career. We can (again very loosely) map these three categories of individual against the two axes of "innovation to create new knowledge", and "skilful application of existing knowledge".

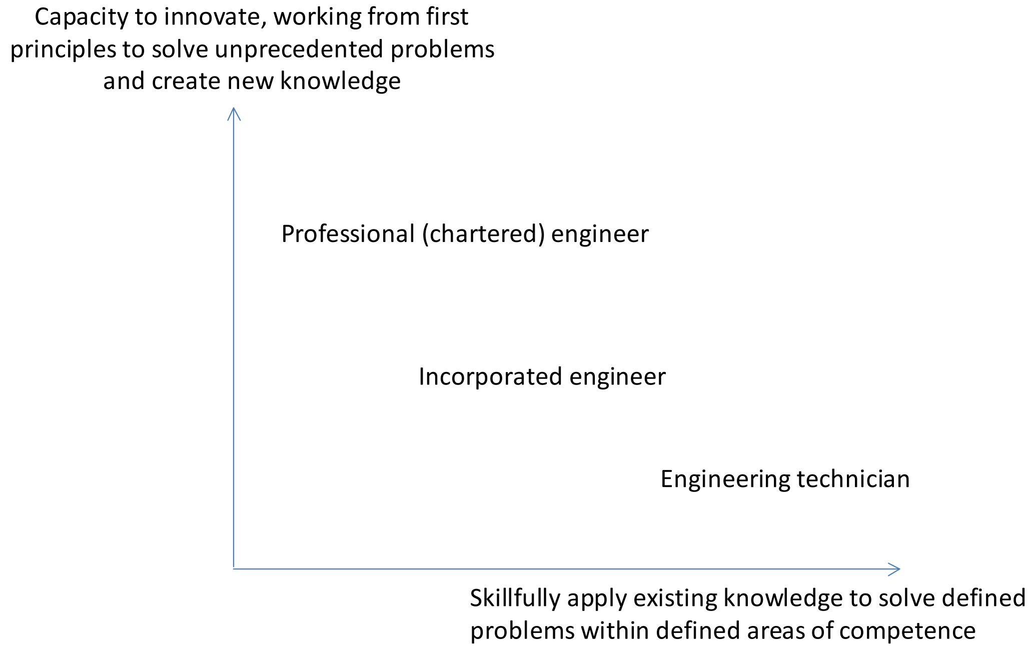

Figure 7: The UK SPEC levels require different balance of Innovation and Skill

The general approach is that the professional level indicates the sorts of roles the person might perform, while the academic qualifications and the institution or professional society memberships indicate the domain of expertise. Engineers are expected to “undertake only professional tasks for which they are competent” and “disclose relevant limitations of competence.” (ECUK 2010)

A key message from the established engineering community is that “anyone calling themselves an engineer must be an engineer”: so there is an expectation that anyone calling themselves a systems engineer would be an engineer first and foremost, with a bias towards “systems” and the ability to apply systems thinking and a systems approach to their engineering domain. This means that a different term would be required for members of other professions applying a systems approach outside engineering.

So how do we – or can we - determine who is competent to handle unprecedented problems and disruptive solutions? Systems engineering has been used successfully on unprecedented problems – the space programme, ICBMs, etc. The UK Engineering Council advises us to “use professional judgement and experience - - judgement - - - to match the nature of the hazard and the level of risk”. Creative and innovative design requires the ability to draw analogies from similar styles of problems, to identify risk areas and define appropriate analyses. Indeed we could describe this as a “creative systems thinking approach”- see for example BKCASE SEBOK Chapter 2 (BKCASE 2010) and (Godfrey et al, 2010).

## “What do Systems Engineers do” versus the wider collaborative landscape that needs Systems Thinking/Systems approach

The root cause of many systems failures is outside the engineering domain; and there is a widespread belief among systems engineers that “the SE process” has value to the “systems approach” outside the boundaries of engineering. How do we reconcile these views with the professional boundaries outlined in the previous section?

It is widely held (Elliott et al, Kasser, - - ) that many major systems projects under-achieve – fail technically, or meet spec but do not deliver value – not because of specifics but because of a failure to manage the overall complexity of the situation. This leads to an increasing interest in the concept of professionals who can assess a high level systems problem and take a global systems approach to the solution – the Stupples level 3, the Hitchins/Kasser level 5. The University of Bristol has an organisation called “the systems centre” – NOT “the systems engineering centre” – because of a belief that there is a need to work across the boundaries of engineering and seek ways of working synergistically across a wider range of disciplines: see the following figure which suggests that there is a role of “systems engineer” as an actor within a much wider community doing systems engineering and using a systems approach.

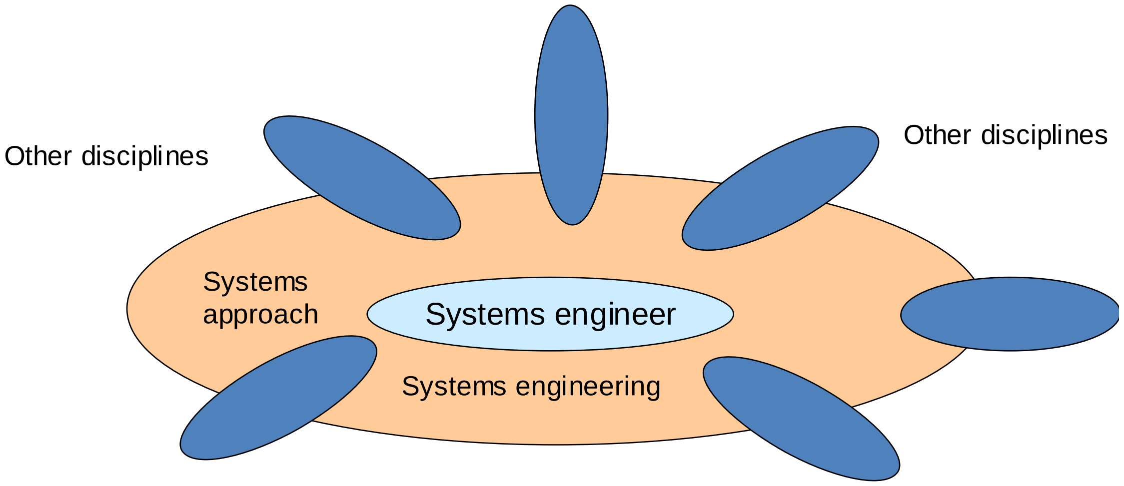

Figure 8: The systems engineer is one actor within a much wider community

It is clear that “systems thinking” is not the sole preserve of systems engineers, and it is also clear that a “systems approach” needs to involve others besides systems engineers. Many people in senior technical, programme and product management positions have been “systems engineers” at some point of their career and now apply what they learnt as systems engineers in a wider context.  Systems engineering as currently formulated tends to focus on product systems. BKCASE explicitly broadens the scope to service and enterprise systems and by considering a “systems approach” abstracted from any domain context. The key issue seems to be that many of the failure modes are due to soft issues not well addressed by current SE standards or published methods (Sillitto 2010). The question is whether SE as “an interdisciplinary approach and means to realise successful systems” can be given a meaning outside the scope of engineering, and can be extended to cover the softer and non-technical issues.

## *Fixing the interface with programme management*

(Rechtin) pointed out that you can’t simultaneously fix cost, time and performance in a situation with high levels of uncertainty.

Yet it is common programme management practice to seek certainty by dividing the job into ever smaller parts, and increase planning time (and cost) in order to increase precision of estimates. If these are not mutually independent and internally highly cohesive, reducing the job into smaller and smaller chunks increases interactions by the square of the number of chunks. So we have a more precise (but not necessarily more accurate) estimate of cost of making the chunks, and more risk through the complexity of the interactions that are not properly managed through the programme lens of cost and schedule.

Commercial practice by contrast is to develop as fast as possible, with lots of feedback and value-based measurement, and (in Toyota anyway) to have a single Chief Engineer who runs the programme and the engineering to achieve product success. The key metric is “time to value”. An effective approach is to focus on increasing shared knowledge, building consensus and discharging risks - “learning by DOING” rather than faith-based PLANNING. Not to say that planning doesn't matter: but the flexibility of the plan and architecture must match the current level of uncertainty and change in the problem/solution space. And decisions must be evidence based not faith-based, and made at the "last responsible moment" not just because the schedule/milestone plan says so.

Looking through out-of-focus spectacles to create a simple model that is useful in this discussion, we can identify seven roles in supply side Systems Engineering:

- "thinker"
- "architect"
- "analyst"
- "engineering manager"
- "requirements engineer"
- "interface manager"
- "IVV manager".

Career incentives and programme pressures pull people into "engineering management" and "requirements" roles, and out of "thinking", "analysis" and "interface engineering".

An option for "systems engineers" as a profession or job family would be to stop overlapping with PM and retreat into the space that is unambiguously technical, making a clear distinction between "SE" and "engineering management"; and to insist on meaningful evidence based measurement of uncertainty and ambiguity. Patrick Godfrey and David Blockley in the UK (Blockley and Godfrey 2000) are doing interesting work on "soft and hard" systems and have a neat concept called the "Italian Flag" – green is what we have evidence works, red is where we have evidence of problems, white is unknown. Note the use of the "E" word, the same as the "E" in Feasibility Evidence Descriptions (Boehm & Lane): opinions and powerpoint do not constitute evidence. Proper analysis and experimentation do, preferably validated by coming at the same issue from several different directions.

An alternative approach is to advocate the Toyota model of the empowered Chief Engineer.

There is also evidence that competitive and transactional behaviours destroy synergy and increase risk, whereas alignment of purpose leads to collaborative behaviours that reduce risk and increase prospects of success – even within time and budget!

## Deductions and Conclusions

There is a clear place for systems engineering and systems engineers within large projects developing complex technology-intensive product systems. There is an increasing demand for systems engineers skilled at handling the system/software interface within such projects.

There is a much wider scope of activity that needs a “systems approach” – including many roles in organisations developing and using high-tech complex systems, products and services. This needs to focus on soft issues and sustainability considerations that are now the limiting factor in most complex system developments, and would ideally be based on an underpinning “systems science” foundation.

Organisations that employ “systems engineers” are in a good position to apply systems engineering to their whole organisation because they have a continuous stream of systems engineers coming through the ranks, some of whom will show aptitude for applying the systems approach in the wider organisation and will know the organisation well enough to be able to do this effectively.

Systems engineering has an image problem outside its traditional domains of application because the use of the word “engineering” leads non-engineers to assume that systems engineering is not for them. Also, the experience, ability and professional level of systems engineers are widely variable, just like any other profession, and not all systems engineers can do all aspects of systems engineering.

Mature disciplines and mature professions segment and specialise to deal with such issues. Physicists specialise in different areas of physics. Doctors specialise in different areas of medicine. The BKCASE SEBOK provides a good basis for defining the overall scope of SE, and GRCSE proposes content for a core element of domain-independent SE courses.

To move forward, the systems engineering community now needs to deliver excellence within the areas that are well defined, and engage synergistically with other communities to foster the systems approach in areas where a systems approach appears to be needed but is not currently used.

## References

ATHERTON J S (2010) *Learning and Teaching; Bloom's taxonomy* [On-line] UK: Available: http://www.learningandteaching.info/learning/bloomtax.htm Accessed: 3 November 2010

ANDERSON, L W, & KRATHWOHL D R (eds.) (2001). A Taxonomy for Learning, Teaching, and Assessing: A Revision of Bloom's Taxonomy of Educational Objectives. New York: Longman

BKCASE 2010: Systems Engineering Body of Knowledge version 0.25, Private Review Version, Stevens Institute of Technology, September 15 2010

BLOOM B S (ed.) (1956) Taxonomy of Educational Objectives, the classification of educational goals – Handbook I: Cognitive Domain New York: McKay

Boehm, B., and Lane, J. (2007) Using the Incremental Commitment Model to Integrate System acquisition, Systems Engineering, and Software Engineering, CrossTalk, October 2007, pp. 4--9.

ECUK: Specification for Professional Engineering Competencies (UK Spec), Engineering Council (UK) 2010

Elliott et al: "Creating systems that work – principles of engineering systems for the 21st century", Royal Academy of Engineering, 2007

Godfrey and Blockley, "doing it differently – systems for rethinking construction", Thomas Telford Press, 2000

Godfrey et al, "What is systems thinking", One-page guide Z-7, Issue 1.0, INCOSE UK, March 2010

Head, Peter: Entering the ecological age: the engineers' role; Institute of Civil Engineers, Brunel lecture series, 2008

Hitchins, Derek, 1995 – "World class systems engineering in the UK", Euroforum conference "managing systems engineering for competitive advantage", Kensington Close Hotel, London, 28-29 June 1995

Hitchins & Kasser, 2009 – paper at APCOSE, Asia Pacific Conference on systems Engineering, Singapore, July 2009 – title to be confirmed

Kasser – private conversations and conference discussions

INCOSE 2010: SE Competencies Framework 2010-0205 Issue 3 UK, INCOSE Technical Product

ISO 2008: ISO/IEC 15288:2008, System lifecycle processes

Presland, Ian: SE competency model - Internal Thales document, 2008

Price and Beasley: Keynote address at INCOSE UK Annual Systems Engineering Conference, 2010

Ring, J., Garcia, J., Yonemoto, K. Modeling a Systems Engineering Enterprise, Conference on Systems Engineering Research, USC, 2008, Paper #177, pg. 3.

Ring, J., "*Engineering Value-seeking Systems*." IEEE-SMC Conference Proceedings, 1998

Rechtin, "Systems Architecting – creating and building complex systems", Prentice Hall, 1991

Sillitto et al, 2009: SEASON report 2009 – systems engineering annual state of the nation, INCOSE UK, February 2009

Sillitto, 2009: "Systems Engineering in context – a thinkpiece" – article in INCOSE UK "Preview" newsletter, Early Summer 2009 edition

Sillitto, 2010: "Design principles for ultra-large scale systems", INCOSE International Symposium, Chicago, July 2010

Stevens, Robert: Presentation at EA Conference, February 2008, London

Warfield – an introduction to systems science, World Scientific, 2006

## Bibligraphy

Hillary Sillitto is Thales UK’s Systems Engineering Director. He started engineering in the 1970s, working on the design of novel optical systems for a wide range of system applications. He then worked successfully on increasingly diverse and complex systems, mostly for defence and aerospace applications, and has held increasingly responsible positions in industry and government, running the UK MOD Integration Authority as an industrial secondee from 2005-8. He has been awarded 9 patents, and has published several papers at INCOSE international symposia (two of them winning Best Paper awards). He is a Visiting Professor at the University of Bristol, and a member of the BKCASE SEBOK author team. He was elected an INCOSE Fellow in 2009 and certified as an ESEP in February 2010. He is also a Chartered Engineer and a Fellow of the Institute of Physics.