The Royal Academy of Engineering

Philosophy of Engineering

Volume 1 of the proceedings of a series of seminars held at The Royal Academy of Engineering

The Royal Academy of Engineering

Philosophy of Engineering

Volume 1 of the proceedings of a series of seminars held at The Royal Academy of Engineering

ISBN 1-903496-38-1

June 2010

Published by
The Royal Academy of Engineering
3 Carlton House Terrace
London
SW1Y 5DG
Tel 020 7766 0600
Fax 020 7930 1549

Copies of this report are available online at:
www.raeng.org.uk/philosophyofengineering

Registered Charity Number: 293074

Table of contents

Foreword

Dr Keith Guy FREng

The Royal Academy of Engineering wants to move engineering to the centre of society by highlighting the crucial role that engineering plays in shaping our lifestyle and culture. The contribution that engineering has made to intellectual history is central to this role. Engineering has had an enormous impact in developing tangible benefits from the complex body of knowledge that humanity has developed. Relativity theory and Darwinian natural selection might be cited as pinnacles in the ever-progressing ascent of human knowledge, but we should add the development of the World Wide Web and space exploration as examples of the awe-inspiring level of understanding that has been reached. Engineering has made an overwhelming contribution to our understanding of the way the world works and how to make the world work for us.

One of the main aims of this seminar series was to develop an appreciation of the nature and role of engineering knowledge. There is great value in developing a better understanding the nature of engineering knowledge and of engineering itself. This allows us to raise the profile of engineering by demonstrating its role in developing sophisticated knowledge. It can also bring a sharper understanding of engineering method, which can be of great value to engineering education. It can also enable a clear formulation of what engineering *is*, in order better to convey its value through public engagement.

As well as a better appreciation of the nature of engineering, the series aims to show that there is much fuel for philosophers if they look to engineering for examples. Philosophers may find in engineering enlightenment on the kinds of questions that they have struggled with for centuries and no doubt philosophers will also find new issues to engage them. Engineering work on artificial intelligence and information technology can, for example, enlighten the philosopher's questions about the nature of thought, consciousness and language. The engineering process of synthesis and construction can inform metaphysical questions about what the world is made of, how it can be broken down and what its fundamental elements are.

There is, however, no point in engaging in a philosophy of engineering unless it has a use - no engineer embarks on a project unless there is an end purpose for what they are working on. The objective of this series is to demonstrate the complexity and richness of engineering and the extent of its influence on human progress. It can be used to send out the message to society that engineering is an important, rewarding and worthwhile profession. In addition, the skills of philosophers in constructing and delivering clear arguments could be of great use to engineers. If philosophy of engineering can help to cultivate such skills in engineers, then engineers will have a stronger voice with which to convey that message.

Engineering is a broad, interdisciplinary field and has links with the social sciences and humanities as well as the natural sciences. The basic aim of the seminars was simply to get engineers and philosophers together to share ideas and to identify research areas of common interest. The Royal Academy of Engineering hopes that this will be the beginning of a fruitful collaboration and that this, and the forthcoming second volume on philosophy of engineering articles, will provide food for thought for philosophers and engineers alike.

Dr Keith Guy FREng
Chair of the philosophy of engineering steering group

Introduction

The papers in this volume are from the first three of a series of seminars in the philosophy of engineering organised by The Royal Academy of Engineering held in 2006 and 2007. The papers were given by philosophers and engineers with an interest in the philosophy of engineering who were invited to speak on a given theme, as described below. The results are a series of papers of great variety, showing a range of different perspectives on the philosophical issues that engineering raises. It is hoped that these will spark philosophical interest and will lead to further, deeper consideration of the topics explored in the seminars.

What is Engineering and What is Engineering Knowledge?

The following questions were offered for consideration at the first seminar: what are the intellectual foundations of engineering? What is engineering knowledge, and what is it to have engineering knowledge? What can philosophers learn from engineering about knowledge? Do engineering and science share a common goal in the quest for knowledge, and do they make equal contributions to our knowledge of the natural world?

In 'Engineering and Truth' Peter Lipton explores the nature of engineering knowledge in comparison with philosophical accounts of scientific knowledge. His question is how engineering differs from science and whether philosophical debates concerning science extend naturally to engineering. Philosophers have long discussed whether we should take scientific theories to be aiming at literal, descriptive truth about the world or whether they should be considered merely as instruments for making accurate predictions. At first sight, engineers may have no interest in the potential truth of theories, since they use them purely as a means to their own practical ends. However, Lipton argues that the real situation may be more complex. Surely an engineer will want to know that the theory is accurate and reliable if that theory is employed in the design of some artifact, and what better guarantee of reliability is there than a theory's truth? As Lipton concludes, there is certainly a need for more exploration of the relation between engineering knowledge and philosophical accounts of science. Science and engineering are close cousins, despite any differences between them, so any philosophical account of science that cannot extend, in adapted form at least, to engineering will be lacking.

When philosophers ask questions about knowledge, especially the fundamental, sceptical questions, they do not ask what it is that we know, but how, if at all, we can know what we know. This question is posed in a more practical form in Sir Tony Hoare's paper 'The Logic of Engineering Design.' Quite often computer programmers do not know how their own programmes work or how they will perform in certain circumstances. Sir Tony argues that this is a situation that can and should be avoided. His view is that if software engineering is carried out on a rigorous scientific basis, then programmers will have a way of knowing how their programmes will function and, more importantly, they will have a way of demonstrating that they will so function. Therefore they can have demonstrable knowledge of what they know. Sir Tony argues that this knowledge can be gained by using the methods of propositional logic - that the conformity of an engineering design to a specification can, in principle, be established by a basic proof in propositional logic. His view is that computer science, as it matures, will come to rest on such perspicuous foundations.

In 'Plato and the Internet: Liberating Knowledge from our Heads', Kieron O'Hara argues that traditional philosophical concerns about knowledge focus only on a limited range of the forms that knowledge takes. Philosophers tend to think of knowledge as having the form of beliefs that an individual can entertain in their mind or write down and pass on. Philosophers question whether and when such beliefs can be considered knowledge. However, O'Hara points out that often engineering knowledge does not take this form, but differs in two important ways. Firstly, some engineering knowledge is a matter of 'know how' - a matter of having a skill or ability. Know how does not take the form of belief and it is not something that can be written down or otherwise formulated in words. Secondly, a great deal of knowledge, especially engineering knowledge, is knowledge possessed by an organisation. It is shared by the members of an organisation, or stored in records and on databases. Therefore, it is not knowledge that is in the mind of an individual. O'Hara argues that computers and the Internet, products of engineering, allow a great deal of knowledge to be shared and 'outsourced' in this way. This shakes up the traditional philosophical picture of knowledge. O'Hara argues that philosophers should turn their attention to such forms of knowledge and apply their skills to dealing with the practical problems of using effectively the knowledge held by organisations.

Systems Engineering and Engineering Design

The second seminar focussed on the nature of engineering design, especially the design of complex systems. Questions for the seminar included the nature of design; design and aesthetics - should engineers prioritise function over form?; the nature of complex systems; the place of human agents in complex engineering projects and the nature of engineering processes.

A significant theme of this set of papers is the importance of considering society when engaged in engineering design. John Turnbull argues in 'The Context and Nature of Engineering Design' that the design engineer's work has an enormous impact on society. Design needs to be focussed on needs and aspirations of people - therefore designers must take into account not only technical details of function and economical concerns but aesthetics and the impact of a product, structure or process on society, including imposed risks. Societal needs and concerns differ across the world and the engineer must be flexible and sensitive to such contexts. Turnbull argues that engineers should play a more active role in society and politics because of the enormous impact they have on the way we live.

Chris Elliott, however, questions the degree to which societal issues are central to engineering design, claiming that technological concerns are always going to be most important to engineering. After all, it is a matter of technical detail rather than social consideration whether a plane has been constructed in such a way that it will fly safely. However, he acknowledges that public interests cannot be ignored because, at the very least, there has to be a market for the products that an engineer designs. In 'Engineering as Synthesis' he argues that the method for successful design in engineering is to think in terms of *integrated system design* - thinking in terms of a whole system and its context. In order to do this successfully the engineer cannot cling to disciplines and specialisations but must be willing to engage in all aspects of a design task.

In Elliott's view, engineering is to be defined in terms of design - that is, the characteristic activity of engineering, which distinguishes it from science and from trades or crafts, is that engineers engage in design. David Andrews concurs with this analysis, and in 'Philosophical Aspects of Engineering Design' he draws out the philosophical issues in the nature of design, arguing the case for the need for a philosophy of engineering design. Indeed, if design is the essence of engineering, a philosophy of design will be essential to a philosophy of engineering. Andrews' paper demonstrates the complexity of engineering design, highlighting the many components in each stage of the design spiral. However, he points out that the most philosophically interesting issue is how the design process is instigated in the first place. The source of inspiration for design ideas and the 'wicked problem' of establishing a design specification from a client's requirements are, he argues, issues that would benefit from philosophical reflection.

The process of engineering design is made more complex when it concerns, as perhaps most engineering designs do, systems that involve both people and machines. In 'Roles and Rules and the Modelling of Socio-Technical Systems' Maarten Franssen discusses the philosophical issues that socio-technical systems give rise to. Franssen asks: how do engineers deal with the fact that some elements of such a system function in accordance with the laws of physics and mathematics, whereas others function according to seemingly different laws? Should the engineer treat people as mechanistic devices, assuming that certain inputs will deterministically yield predictable outputs, or should engineers look to the social sciences for alternative laws for modelling the human elements of such systems? These questions raise both deep philosophical questions about the possibility of a unified description and set of rules for a socio-technical system and pressing practical problems for how to design such systems so that they are functional and safe.

AI and IT: Where Engineering and Philosophy Meet

The development of artificial intelligence and the growth of information technologies involve engineering activities that are probably closest to the interests and practice of philosophy. The questions set for this seminar included the following: what can engineering tell us about consciousness? Who decides when we have built a conscious machine - the philosopher or the engineer? What can philosophers and engineers contribute *together* to the study of, and research into, AI? What issues does the Web, especially the Semantic Web, create for philosophy?

The actual building of a conscious machine or robot is a task for engineers, but it is essentially tied to understanding the nature of consciousness. Igor Aleksander, in 'The Engineering of Phenomenological Systems', argues that to successfully construct a machine with phenomenological awareness it is necessary to turn to philosophy to gain a better understanding of the goal of that project. However, he demonstrates that engaging in the engineering project itself affords insight on the philosophical questions about consciousness. Aleksander argues that an important aspect of building a conscious machine is building a physical machine that can move like a person. An essential aspect of being conscious of the outside world is reflexive awareness of being a thing in the world - and this is inherently related to being a physical thing situated in a particular environment. Philosophers considering the nature of consciousness have in some cases failed to fully appreciate the importance of the physical body to consciousness, so this work by engineers and scientists on the nature of consciousness delivers real philosophical insight.

Ron Chrisley, in 'Interactive Empiricism: The Philosopher in the Machine' reiterates the idea that through the process of engineering we can gain philosophical enlightenment. He argues that sometimes conceptual change is not brought about through explanations or argument, but through activity or interaction. Just as a description or explanation of how to ride a bicycle cannot by itself convey that skill, developing a philosophical understanding of consciousness may require engagement in practical activity in addition to philosophical considerations. The attempts to construct conscious systems described by Aleksander are the kinds of activities that can bring such insight. Chrisley uses this example as evidence that some kinds of philosophical understanding are not gained through either of the two traditional paths of rational reflection or a search for empirical evidence, but by a third way - by interaction with systems through attempts to build, fix or modify them. Is this a case for arguing that engineering knowledge is of a novel form, creating knowledge where data collection or theorising do not deliver?

This is the first volume of papers from the philosophy of engineering seminar series. Topics to follow are engineering and metaphysics, engineering and environmental ethics and philosophy in engineering education.

Part I: What is engineering and what is engineering knowledge?

1. Engineering and Truth

Professor Peter Lipton
Department of History and Philosophy of Science
Free School Lane, Cambridge CB2 3RH

Peter Lipton was the Hans Rausing Professor and Head of the Department of the History and Philosophy of Science at the University of Cambridge and a Fellow of King's College. His philosophical interests included the structures of explanation and inference in science, the nature of scientific progress, social epistemology, the relation between science and religion, and biomedical ethics.

Peter Lipton passed away on 25 November 2007. He is greatly missed.

What do philosophers of science think they are doing?

I am a philosopher of science: what do I do? Here is the short version: astronomers study the galaxies; I study the astronomers. But what is the point of doing that? It has been said that scientists need the philosophy of science like birds need ornithology. That is close to my own view. Philosophy of science, along with the history and sociology of science, is about how science works and what it should be taken to achieve, but one should be relatively modest about the prospects for turning the philosophy of science into a kind of technology for improving scientific practice. If you had to justify the philosophy of science purely in terms of the direct aid it could give to scientists and engineers in improving their day-to-day work, you would have trouble getting a grant - but I do not think you have to justify the philosophy of science entirely or even primarily in those terms. Science and engineering are among the most impressive activities that we have engaged in as a species, so it is well worth a few philosophers' time trying to understand a little better how those activities actually work, even if that understanding does not translate directly into improved performance, just as it is well worth a few astronomers' time trying to understand a little better how galaxies work, even though the astronomers do not hope to improve galactic behaviour.

But if philosophers are just out to describe how science and engineering work, why do they make such a palaver out of it? Why not just ask a friendly practitioner what she does? There are several reasons why scientists and engineers do not already know all the answers that philosophers are looking for. Philosophers are not only interested in description: they are also interested in explaining why the practitioners do what they do, and also in questions of justification, questions such as whether we are ever entitled to believe that a high-level scientific theory is even close to the truth. These are not the sorts of questions to which practitioners need have a ready answer. Moreover, even when it comes to the question simply of describing how scientists and engineers actually go about their business, there is no reason to believe that they will be particularly good at telling us. Practitioners are in the business of doing, not of describing what they do and there is quite generally an enormous contrast between what we are good at doing and what we are good at describing. It is one thing to be good at riding a bicycle; it is something entirely different to be any good at describing the physics and the physiology of bicycle riding. It is one thing to be very good at distinguishing grammatical from ungrammatical strings of words in your own language, but something entirely different to be able to specify the principles by which those discriminations are made.

It is the same with doing science and engineering. The practitioners can be very good at doing what they do, but very poor at describing how they do it. They do not need to describe it in order to do it. Even more surprisingly, they do not even need to be very good at describing what they do in order to teach others how to do it, because teaching often works in different ways, by example and by encouraging imitation, rather than by giving detailed or general descriptions. (I will return to this point below.) Philosophers of science are in a different position. They need to give descriptions, because giving descriptions of science and engineering is their job. We are not doing the science and we are not doing the engineering, but we are trying to give some sort of general description of what is going on there. This is primarily, in my view, just in order better to understand this incredibly significant human activity. If anything we come up with can improve the activity, all the better; but I do not want to pin my salary on that promise. After all, all we are doing is coming up with descriptions. Those descriptions are surprisingly limited - what we are describing being so complex and subtle - and even if they were accurate, following descriptions is not the way scientists and engineers learn their craft.

In addition to the surprisingly difficult problem of description, we philosophers have, as I have already mentioned, the problem of justification. To put it in its crudest form, the question here is whether scientists really know what they are talking about. Is science really in the truth business, disclosing the nature of a largely invisible reality? Or should we

understand the aims and achievements of science differently? As in most areas of epistemology, or the theory of knowledge, much of the philosophical work on the justificatory questions in the epistemology of science is driven by sceptical arguments which purport to show, often with annoyingly impressive force, that we don't know what we think we know. Many of these arguments are based on problems of under-determination. These are problems of extreme extrapolation. The issue can be developed from either a logical or a physical point of view. From a logical point of view, the problem is that all our theories in science and engineering go massively beyond the data that support them. Indeed that is the name of the game: theories would be of scant interest if they simply summarised what has already been seen. But because they go beyond the data that supports them, all those data are compatible not just with the theories we endorse, but also with many, many other theories - most of which we have not even considered - that are logically incompatible with the endorsed theories, but all compatible with all the same available data. The problem for the philosopher and, to some extent, for the practising scientist and engineer, is how to justify a preference for one hypothesis or theory over another, when both hypotheses are compatible with the same evidence. It is difficult to see how to do this; but if it can't be done then we seem to have no reason to think that our favoured theories are correct rather than any of the innumerable possible alternatives. Maybe we prefer one theory not just because it fits the data, but because it is simple or beautiful, whereas many of the alternatives would be complicated or ugly. But how can one justify the claim that the world is likely to be simple or beautiful?

To see what is essentially the same problem but from a causal rather than a logical point of view, think about the enormous scope of the claims made by scientists and engineers. Think, for example, about the range of claims made about the structure of world in distant places in space and time. Now contrast that enormous range in our assertions with the miniscule surface area where we, as physical creatures, actually touch the world we are describing. Our physical contact with the world we are describing is minute compared to the range of claims we make on the basis of that causal interaction. So, once again, we face massive underdetermination. Hold that interface constant: it is compatible not just with all the stories we tell about what it going on elsewhere in space and time, but with innumerably many other, incompatible stories. On what basis could we be entitled to prefer one such story over another?

Three plausible contrasts

Although there is a healthy sub-discipline of the philosophy of engineering, most of the work in the larger discipline of the philosophy of science has focussed almost exclusively on pure science, and indeed often just on physics rather than on the other sciences. For that reason, and because I am by formation a philosopher of science rather than a philosopher of engineering, I would like to engage in some 'compare and contrast'. Given how general and abstract the issues in the philosophy of science are, surely many of them will apply to engineering too. But I am here particularly interested in where the difference makes a difference, where the philosophical issues change if we focus on engineering rather than on physics. I do not suppose that there is a clear demarcation between science and engineering, but it does seem to me that there are real and relevant differences between pure and applied work that are of philosophical import. And here I want to flag just three candidate differences between science and engineering: a different in output, a difference in knowledge, and a difference in drivers.

As many philosophers of science would have it, the ultimate output in science is theory - a set of propositions, a set of equations, a set of assertions. Perhaps that is a defensible view of science, but it certainly does not seem to do justice to engineering. Of course you can't do engineering without generating propositions, for example in specifying a design, but the ultimate output is an artefact, not a statement. It is something physical and manufactured. And one might expect that this contrast in ultimate output should make a difference to the form that a proper philosophical analysis should take.

The second contrast concerns knowledge. Epistemologists are fond of a distinction between 'knowing that' and 'knowing how'. Knowing that is propositional knowledge - it is knowing that something is the case. It is knowing that a statement is true or that a hypothesis is correct. By contrast, knowing how is an ability or skill. These two kinds of knowledge seem quite different. It was, for example, that difference that I exploited near the beginning of this essay, when I contrasted scientists' know how with their inability to describe those skills, an absence of knowing that. Knowing how does not entail the corresponding knowing that. Nor does knowing that entail the corresponding know how. The contrast between knowing that and knowing how suggests a contrast between science and engineering that parallels my first contrast between theory and artefact. Philosophers investigating scientific knowledge have concentrated on knowledge that; but if we want to do justice to engineering, it would appear that we need to put considerably more weight onto know how.

My third contrast has a different character. It is motivated by the thought that the drivers for problem choice in science and in engineering may tend to be different in kind. In pure science, the driver is often internal to the scientific community: scientists often get to choose their own problems. In engineering, by contrast, it is more common for the driver to be external to the community of practitioners. Engineers often do not get to choose their own problems, but have them chosen instead by government, by industry or by other external sources. Here again, we have a difference that may make a difference, since the way problems are selected may make a difference to the way they are addressed, such that a philosophical account that is more or less suitable to science does not as it stands do justice to the realities of engineering practice. We will come back to see how this might be so.

Those are my three plausible contrasts between science and engineering. There is the contrast in output between theory and artefact, the contrast in knowledge type between knowing that and knowing how, and the contrast between internal and external drivers for problem selection. All three may make a difference to the kind of philosophical analysis required.

Karl Popper

I want now to pursue the compare and contrast strategy from a different angle, by looking at some of the distinctive claims of two of the most important twentieth century figures in the history and philosophy of science: Karl Popper and Thomas Kuhn. There are two features of Popper's position that I will flag. The first is the sharp contrast he drew between the mechanism by which hypotheses are generated and the mechanism by which hypotheses are subsequently tested. His view was these mechanisms are quite distinct, and that the philosopher should only be concerned with the second one. According to Popper, the philosophy of science should be the logic of testing. For him, where hypotheses come from is not a philosophical question. (Thus the English title of Popper's classic book - *The Logic of Scientific Discovery* - is a misnomer, since one of his central claims is that there is no logic of discovery, only of testing [1].)

The other feature that I will emphasise is at the core of Popper's philosophy. One of the reasons I like Popper so much is that he was a philosophical hedgehog. According to the ancient Greek saying, there are foxes and hedgehogs. The fox has many ideas; but the hedgehog has one big idea. Popper is one of philosophy's hedgehogs. He had one big idea and he made the most out of it. His big idea is the asymmetry between good news and bad news from the data, between passing an experimental test and failing an experimental test. His point was that, at least in the simplest cases, one piece of negative evidence refutes a hypothesis, whereas no amount of positive evidence verifies a hypothesis. If your hypothesis is that all swans are white, then no matter how many white swans you see, you will never prove your hypothesis to be correct. No matter how many white swans you see, it is always possible that, locked away in some closet somewhere there is a non-white swan. If such a swan exists, then your hypothesis is false, even though all the data that you had seemed to confirm the hypothesis. By contrast, if you ever do see a counterexample to your hypothesis, say on a visit to Australia, then you have refuted the hypothesis that all swans are white, because the existence of that one counter-example is incompatible with the general hypothesis. Even if every future swan is white, the hypothesis that all swans are white is false.

The methodological moral that Popper drew from this logical asymmetry is that the only relevant evidence is negative evidence. All that scientists should be trying to do is to refute their hypotheses. They should not be looking for good news, but should always be trying to find out what is wrong with their ideas. When they do, they should replace them with other ideas that have not yet been refuted, and then try to refute those. This is an endless process but Popper thought that it was the only thing to do if you are interested in the truth: to construct refutable hypotheses and then attempt to refute them. If by contrast you make a claim which could not be refuted, where there could be no possible bad news, that just means - according to Popper - that you are no longer doing empirical science. To be a scientist is to make a falsifiable claim - it is to say something that could be refuted by empirical evidence. Of course Popper was not saying that a true hypothesis would thereby be unscientific. So far as I know, the hypothesis that all ravens are black is true, yet it still counts as scientific for Popper, insofar as one could say what would count as observing a non-black raven. If, by contrast, 'raven' is defined so as to entail being black, then no counterexample would be possible and the hypothesis would be reduced from being an empirical and scientific claim to being a definition.

Both of these Popperian ideas - the distinction between hypothesis generation and hypothesis testing, and the asymmetry between positive and negative evidence - face at least two particular difficulties in the context of engineering. The intended effect of Popper's distinction between generation and testing is the deflect the philosophers interest away from the process of hypothesis generation. This is to neglect a crucial aspect of research. We do not just want to try to describe how ideas are tested, but we also want to try to say something about how ideas are generated. Popper was very fond of a picture of scientific evolution modelled on natural selection. That analogy may be genuinely

helpful in some respects, but the idea that hypothesis generation is like random mutation is potentially misleading. Hypothesis generation is highly constrained and directed, and would have to be if science is possible at all. Moreover, although here I am venturing into an area where the readers of this essay will be far more knowledgeable than its author, in engineering the structured nature of hypothesis generation may be particularly pronounced, because of a focus on explicit and articulated methodologies of design.

Popper's second idea of testing as falsification has also been criticised in the context of pure science, and this too may face special additional difficulties in the context of engineering. Popper has been criticised for making it sound as though hypotheses are much easier to refute than they really are. Nevertheless, his logical point is sound. If your hypothesis is of the simple form 'All Fs are G' and there is an F that is not-G, then the hypothesis must be false. A counterexample refutes the generalisation. But in engineering, the objects of test are not just general hypotheses; they are also particular artefacts. And the fact that a particular artefact fails a particular test at a particular time does not entail that it will fail on other occasions or that other, similar artefacts will fail similar tests. Engineers are not just interested in whether general hypotheses are correct, but also in whether particular applications will work.

This focus on application brings out another striking weakness in Popper's account, which arises even for general hypotheses when what is at issue is the application rather than the testing. For to use a general hypothesis in an engineering context we need to have some way of judging how likely it is that the predictions derived from it will be correct; but the Popperian scheme is completely unhelpful on this score. If a counterexample to a hypothesis is found, that tells us that the hypothesis must be false; but Popper does not allow that it tells us anything about the next instance. Even more remarkably, no matter how many tests a hypothesis has passed, this gives no information about how likely it is to succeed in its next application. Popper says that a hypothesis that has passed many severe tests thereby becomes highly 'corroborated'; but according to him that is simply a report of past performance that has no predictive value whatever. A strictly Popperian account cannot make sense of applied science. Popper has usefully emphasised the methodological importance of negative evidence, but we need a more nuanced model to account properly for its force.

Thomas Kuhn

The other great figure in 20th century science studies was Thomas Kuhn. Although this is less immediately apparent than in the case of Popper, Kuhn too was a hedgehog: he too had one big idea, which he developed especially in his seminal book *The Structure of Scientific Revolutions* [2]. (This book title is also misleading, since he claimed that scientific revolutions are essentially unstructured events.) Kuhn's big idea is also a contrast, but it is the contrast between rules for research and what he called exemplars, the core meaning of his notorious and protean term 'paradigm'. How do exemplars differ from rules? Kuhn was struck by a feature of research during periods of what he came to call 'normal science'. Members of the research community in a particular speciality act as if they shared the same rules of research. They tend to agree on which are the important problems, they tend to agree on how those problems should be tackled, and they tend to agree on the appropriate standards for judging whether a particular solution is sound. It is as if all of the people in this community are following the same secret rule book.

The problem with this explanation for the coordination of research activity is that the secret rule book does not exist. This formed Kuhn's central question: how is rule-like behaviour to be explained in the absence of rules? There must exist some kind of functional equivalent for the rules. Kuhn needed to find something that does the job that the rules would have done but, unlike the rules, something that is actually present and available to the scientist. What Kuhn found are exemplars. Exemplars are not rules: they are not general principles. Rather, they are specific, canonical problem solutions in a particular scientific specialty. They are particular solutions, but they nevertheless function as general models because, according to Kuhn, they set up similarity relations, so that the scientists can work by imitation and analogy rather than by rules.

Because of the similarity relations that the exemplars elicit, certain unsolved problems come to look similar to the problems that the exemplars solve. Those are the problems the scientist will pick, the outstanding problems that look like problems that have already been solved, and so the problems for which there seems every possibility of successful solutions. To achieve them, scientists will naturally try to apply techniques similar to those that worked so well in the exemplars. That is only rational: since the problems were selected because of their perceived similarity to problems the scientists have already solved, they will try similar solutions to see whether they work on the new problems. And finally the standards by which the scientists judges the new solutions will be the standards that are embodied in the original exemplars. So, although exemplars differ from rules in their content, they are similar to rules in their functions.

Exemplars are specific in content, but general in import and, according to Kuhn, they structure scientific research, guiding scientists in the selection of new problems, in the construction of possible solutions, and in the evaluation of the solutions proposed.

Kuhn's account of the history of the mature sciences is a history of phases: normal science - crisis - revolution - normal science. This cycle is explained by the exemplar mechanism. In normal science the community has a set of exemplars they share and that supports the puzzle-solving tradition that engages them. That tradition goes into crisis when the exemplars continue to select problems, but no longer seem to support solutions. Finally, a scientific revolution is the replacement of one set of exemplars with a new set and so the creation of a new web of similarity relations and a new normal science tradition.

Does Kuhn's story of the exemplar mechanism apply as well to engineering as it applies to pure science? Interestingly, in an article entitled 'The Essential Tension' in a book of the same title, Kuhn himself suggested not [3]. In so doing, Kuhn inverted common stereotypes of scientists and engineers. The common stereotype of the pure scientist is a kind of Einstein figure, wildly intelligent and unconventional, full of new ideas that come from nowhere. The common stereotype of the engineer is rather different, of someone earnest and hard-working, but much more intellectually conservative. Kuhn proposed that the truth has in a way to be the opposite of this, because it is the pure scientist who can very often afford to be intellectually conservative and it is the engineer who must sometimes be unconventional. The reason the scientist can afford to be conservative is because she gets to choose her own problems. So she can use the exemplar mechanism as a mechanism of problem selection. She can choose new problems which seem to her to be similar to the problem that she has already solved, so she can afford to be conservative in the techniques that she applies to solve those new problems. Engineers, by contrast, have their problems imposed from without, as argued earlier. So they have no reason to suppose that the problems they have to solve will be all that similar to the problems they have already solved. It is therefore the engineer who may have to be more adventurous, and more of an intellectual opportunist. The exemplar mechanism will have less force and power for the engineer, therefore, than for the pure scientist. And we might expect that the dramatic contrast that Kuhn finds between normal and revolutionary science is considerably attenuated in the case of engineering, in part because, if Kuhn's suggestion is along the right lines then 'normal' engineering is more revolutionary and 'revolutionary' engineering more normal than those periods are in pure science.

Fruit and light

I have so far been focussing primarily on various ways science and engineering might differ in their methods of inquiry. The brief tour of the views of Popper and Kuhn suggests that there may be difference both in the roles of negative evidence and of positive models. But now I want to turn from contrasts in method to contrasts in aims or goals. The classic picture of science includes two aims, what Francis Bacon called 'fruit' and 'light'. Fruit is the application of science to enable us to anticipate and to control nature. Light is understanding, seeing how the world works. Philosophers of science are very much occupied by the light question. Is science really in the truth business? Is science really revealing the reality out there, as it is in itself, although it is largely unobservable? The question strikes philosophers as especially pressing in light of the underdetermination problem that I mentioned at the start of this essay: the fact that no amount of data are sufficient to single out a theory uniquely.

Let me briefly relate the two best known philosophical arguments on the truth question - on the question of whether we ought to say that any of our scientific theories are even approximately true. One argument optimistically says yes, the other pessimistically says no. And both arguments are probably unsound as they stand. (Here I am aware that, unlike what Kuhn had to say about scientists and engineers, I may just be reinforcing certain stereotypes about philosophers.) The optimistic argument is highly intuitive. Take our most successful scientific theories. What would explain their remarkable predictive successes? If the theories were true, or approximately true, then the predictions deduced from them would come out true as well. So the truth of the theory would explain its predictive successes. Of course it is logically possible that the theory is enormously predictively successful even though it fundamentally mistaken. The bits you do not check are entirely wrong but every bit you do check just happens to be right. But that would require in Harvard philosopher Hilary Putnam's memorable way of putting it, a miracle, and you should not believe in miracles [4]. Such a extraordinary coincidence is possible but highly unlikely, so what is much more likely is that the theory is at least approximately correct. That is the optimistic argument on the truth question. It would be a miracle if a highly successful theory were not at least approximately true.

But there are difficulties with this argument. One difficulty is that it seems subtly to beg the question in favour of the truth view. Rather unusually for a philosophical argument, the miracle argument is broadly empirical. Philosophers do not normally give empirical arguments; their arguments do not normally depend on observation and experiment. The miracle argument, however, is saying that we ought to infer from the empirical success of a given theory - itself an empirical fact - to its truth. The circularity objection is that this form of reasoning from the success of the theory to its truth on the grounds that the truth would best explain that success, is characteristic reasoning within science (Lipton [5]). Scientists infer that the best explanation of their data is probably correct.

But the issue is precisely whether that form of reasoning is taking us towards the truth. So, if you were doubtful about whether scientific methods are actually taking us towards the truth, you ought to be doubtful about whether this optimistic argument is actually taking us towards the truth. One might admit that the truth of the theory would be the best explanation for its success. Why does that mean that we should believe that the theory is true, unless we are already committed to this form of reasoning? That is the first objection to the miracle argument.

The second objection is not that the miracle argument begs the question, but that it commits what is known as a base-rate fallacy. The fallacy arises because, somewhat surprisingly, while all Fs are G does entail its converse, namely that all not-Gs are not-F, *most* Fs are G does not entail its converse, namely that most not-Gs are not-F. And yet this seems the form of the miracle argument. The key premise is that most false theories would have been predictively unsuccessful - most false theories would have revealed their falsity by failing when we tested them. Now this premise may perhaps be true, but what the argument draws from this simply does not follow, namely that most successful theories are true.

To make this point more vivid, suppose that you buy a lottery ticket with eight digits. You then watch the balls coming up as the numbers are called. The first two balls come up, and your ticket matches both. What can you say at this stage? Well, most tickets would already have failed: only one in a hundred get those two digits right. So most 'false' (that is, losing) tickets, would already have broken down on the first two numbers, which you successfully 'predicted'. But do you infer from this that you are almost certainly quids in, because most tickets that get the first two numbers right hit the jackpot? I hope not, because although your odds of winning have improved, they remain only one in a million, there being still six undetermined digits on your ticket. Most losing tickets get the first two digits wrong, but it manifestly does not follow that most tickets that get the first two digits right are winners. Similarly even if most false theories are unsuccessful, it sadly does not follow that most successful theories are true. All unsuccessful theories are false, but it looks like most successful theories are false too: we just haven't yet seen where they fail.

I turn now from the miracle argument for truth to the pessimistic argument against truth. Like the miracle argument, this 'pessimistic induction' is rather surprisingly an empirical and, in this case, an historical argument. The history of science is a graveyard of putative entities that turn out not to exist, of putative processes that turn out not to take place and of theories that turn out to be false. Almost every scientific theory, more than - choose your own figure - 150 or 200 years old, we now know to be, strictly speaking, false. To put it as crudely as I can, all past theories have been found to be false and therefore it is very likely that all present theories and, probably, all future theories, will eventually be found to be false as well. Our best current theories may look terrific at the moment, but that is because we are stuck in the present. If we step back a little and achieve some historical perspective, we should say that it is just a matter of time before the warts start to show. We will find out that we are wrong now, as we were in the past. That is the pessimistic induction.

One reply is to concede that it is very unlikely that any of our major theories are correct in every respect. But that leaves room for a picture of the history of science in which there are successively improving approximations to the truth, so that later theories tend to be closer to the truth than the earlier theories they replace. And that is all the glory that any scientific realist - anyone who thinks that science is in the truth business - ought to desire. To this the pessimist may respond by denying the realist even a story of improving approximation to the truth, on the grounds that such a story would require that we be able to tell a plausible story about the history of science that exhibits the right kind of directionality. It requires a trajectory that can be seen to move steadily towards a particular view, even if the approach is asymptotic. But the pessimists deny that any such historical curve is plausible. Scientists' claims about the world have not shown that kind of steady development. As Kuhn for example claims, there are too many scientific revolutions, and they haven't taken us in a single direction. Our view of the fundamental nature of reality changes dramatically over the history of science, so there is no entitlement even to the approximation version of the realist view.

Another way of objecting to the pessimistic induction is to point to its deceptively simple form. It has the form all observed Fs have been G, therefore probably the next F will be G. All past theories have been found to be false, therefore present theories will probably turn out to be false as well. However, inferences of this simple form are not always warranted. In particular, they are not warranted if there is reason to believe that the dataset is not a representative sample of the broader population. In that case, even if all of the observed Fs are G, that does not provide a reason to think that the next F will be G. Thus all the ravens you have observed may have been black, but if they have all been British ravens and the prediction concerns Arctic ravens, the inference may be unfounded.

Applied to the pessimistic induction, the question is whether past theories are representative of theories generally. The theories are not the same, they are not independent, and their environments are not the same. Taking a leaf out of Popper's book, we might argue that we have learned from our mistakes and that the fact that we know that the past theories were mistaken does not make it more likely that present and future theories are also mistaken, but rather makes it more likely that present and future theories will be correct, or at least more correct.

How do these arguments for and against truth apply when we focus more single-mindedly on engineering? Should engineers even care about the truth question? In terms of Bacon's distinction between fruit and light, one might be tempted to propose a division of labour. Scientists can worry about the light, the truth question; but engineers just care about the fruit, about practical application, about building something that gets the job done.

On this view, it looks as though the truth debate ought not to worry the engineer; but I am not so sure. Even if you take the view that discovering the way nature really works is not the ultimate aim of your activity, even if you have a more practical aim, you may nevertheless maintain that discovering how nature works is a crucial means to the practical goal. Understanding better how nature really works is getting the truth about nature - so you will end up caring about truth after all. The truth may be valuable to both communities, even if for one it is an end in itself, while to the other it is only a means to another, practical end.

This comparison of attitudes towards truth needs much more work. For even if both communities are interested in the truth, they may not be interested in the same truths. Scientists may care more about underlying entities; engineers more about observable outcomes. And scientists may care more about the description of highly idealised systems; engineers about real-world processes. We need a better understanding of the roles of models and approximations in both areas. There is plenty more work to do in the project of extending the philosophy of science properly into the philosophy of engineering, a project that will reveal more similarities and more differences.

Bibliography

1. Popper, Karl (1959) *The Logic of Scientific Discovery*, London: Hutchinson.
2. Kuhn, Thomas (1970) *The Structure of Scientific Revolutions*, 2nd ed., Chicago: University of Chicago Press.
3. Kuhn, Thomas (1977) *The Essential Tension, Chicago:* University of Chicago Press.
4. Putnam, Hillary (1978) *Meaning and the Moral Sciences*, London: Hutchinson.
5. Lipton, Peter (2004) *Inference to the Best Explanation 2nd edition*, London: Routledge.

2. The Logic of Engineering Design

Professor Sir Tony Hoare FREng FRS
Microsoft Research, Cambridge

Tony Hoare took a degree in Lit. Hum. from Oxford University in 1956. As an undergraduate, he was inspired by his philosophy tutor John Lucas to pursue his interests in symbolic logic and computing. His working career started in 1960 as a computer programmer with a small British computer manufacturer, and since 1999 he has been a computer researcher at Microsoft Research Ltd. Cambridge. In between, he had a distinguished academic career at the Queen's University, Belfast and at Oxford University, and set up the first computing degree programmes in both universities. His long-term research goal has been to place the engineering of software on a firm basis of scientific theory.

In my Greats degree at Oxford University, philosophy was my favourite subject. I was fascinated particularly by the philosophy of mathematics, and wondered at the certainty we ascribe to its proven theorems. I explored the way in which mathematical proof is based on the axioms and deduction rules of modern logic and set theory, and I speculated on the possibility of involving computers in mathematical reasoning. My experience as an engineer dates from 1960, when I started my career as a computer programmer. I greatly enjoyed my first couple of years actually writing programs; I was then promoted as a manager of programmers and then as a research scientist in computing, which I have been ever since. I have always taken an interest in the philosophical aspects of program construction, and now I would like to share with you some of my thoughts on the subject. My main thesis is that computing is a branch of engineering science. As an engineering discipline, it is motivated to build products of widespread practical use to their users. As a science, it is motivated by curiosity to answer basic questions about its subject matter, in particular about computer programs. Like other branches of engineering, it uses logical and mathematical reasoning to guide the evolution of a program design, and to check soundness and correctness before delivery of a software product. These checks are increasingly conducted by the computer itself, before embarking on execution of the program.

I will concentrate attention on the areas in which engineering and science have most in common, and the areas in which science can give the most help to engineers. I will ignore for the time being all the fascinating and wide-ranging differences between science and engineering, and the often conflicting culture and practices of scientists and engineers.

The similarity can be summarised by noting that they ask the same kinds of basic questions about the subject matter of their study, and they differ only in what they ask the questions about. In the case of science, the subject of study is the natural world - the objects and organisms and substances and phenomena that we observe in our immediate or more distant environment. In the case of engineering, the object of study is the artificial world - the products which are designed and made by engineers themselves for the use and benefit of mankind. Ignoring the irrelevant distinction between the natural and artificial world, I will claim that scientists and engineers use the same kinds of mathematical and logical reasoning to draw conclusions from their understanding of the relevant basic science. The scientist uses the conclusions of reasoning to check the consistency of a general theory with the results of a wide range of experiment. The engineer uses them to check the quality of the design of a product, its reliability and its conformity to the needs of its users.

Basic questions of engineering

The first question that an engineer asks about what he is designing and building is: What is it for? In other words, what human requirements does it meet? This very important question is deliberately ignored by modern science, which studies external natural phenomena: the natural world is no longer believed to have any human-oriented purpose. Since I want to concentrate on similarities between engineering and science, I will ignore this first and most important question, and proceed forthwith to the next basic question.

What does the product do? In greater detail: What are its properties and behaviour? How does it interact with its user and its external environment? Here, the engineer often gives an answer more technical in detail than the average user of the product would be interested in. The answer is likely to contain scientific technical terms, like ohms and farads, which the average user will not understand or be interested in. You might think that the engineer who actually designs and implements a product should find this question trivially easy to answer. Surprisingly, in the case of complex systems like a computer program, this is not so. Even people who have just finished writing a program are often quite puzzled about what it actually does: if you ask an awkward question, they will have to conduct an experiment, actually running the program in order to find out what it does. In the ideal, this should not be necessary. A specification of the behaviour of a program can in principle be written in advance, perhaps at the beginning of the project, and its accuracy should be maintained throughout design and implementation.

The third question is one that surely interests all engineers - indeed it probably was the initial motivation for their choice of engineering as the subject for their study. It is: how does the product work? How does the engine actually function, and how does it drive the wheels of a car? How does the aeroplane fly? The answer to this question is usually given by describing the structure of the product and its components. It includes a description of the ways in which the components are connected together, and the methods that they use to interact with each other.

Again, we know that many good software engineers are seriously challenged to answer questions like this about their own programs. A few weeks after writing a program, they no longer know how it works. This causes problems when attempting to diagnose and repair design errors that come to light later. It causes even more severe problems when the need arises to produce the next version of their program, when needed to make it do something a little different or a little better. The programmers then have to find out again by experiment how the current version of the program actually works.

The two questions 'what?' and 'how?' are equally relevant to the pursuit of all branches of natural science. For example, a classificatory biologist may enquire what does a newt or an axolotl do? How does it relate to its environment? And the next question asks how the creature is constructed: what are its limbs and organs, and how do they interact? Sciences which concentrate just on these two questions are often characterised (and sometimes disparaged) as being merely descriptive.

Basic questions of science

The more mature branches of science are certainly based on an extensive foundation of accurate description; but then they go on to address some rather deeper questions. The researcher in engineering science asks the same kind of deeper question. The first of these is: Why does the product work? What are the basic scientific principles, the equations and the laws of nature, on which the working of the product actually depends? So the aeronautical engineer studies aerodynamics, which makes explicit the laws that explain why an aeroplane flies. On the basis of the laws, it is possible to make predictions about how the aeroplane will respond to its controls; the modern engineer exploits such laws to optimise the quality of the product, and to reduce its cost.

The final and most distinctive feature of modern science is its pursuit of certainty of knowledge. The goal of the scientist is to assemble a massive body of convincing evidence that the answers to all the previous questions are in fact correct. Peter Lipton has described Popper's theories of how scientific hypotheses are corroborated by a determined experimental attempt to refute them. The engineer similarly uses a wide range of testing before delivery of a product, to gain confidence that it will not fail after delivery. Testing is also used to detect and remove any remaining deficiencies in the product that appear only after implementation. In architecture, this is known as 'snagging', and in programming it is called 'debugging'. I regret to confess that programmers often spend around half their total professional time in debugging their programs, detecting and removing design errors that have been made earlier by themselves or by their colleagues. Excessive reliance on debugging may explain or even justify the historical reluctance of more traditional engineers and scientists to regard computer programming as an engineering profession. What is worse, the overall costs of programming error to the economy of the world (including precautionary palliative measures) are currently estimated at tens of billions of dollars. Most of this falls on the users of the software rather than the producers.

In a mature engineering discipline, post hoc testing of a product is not the only way of ensuring reliability and conformity to user specification. Once there is a well-established scientific theory, it is possible to calculate in advance of delivery that a product will pass every envisaged test in every reasonably predictable environment that may arise in its practical use. For example, in civil engineering there is an established theory of how a building or a bridge will stand up to stresses and strains caused by high winds. From this underlying scientific theory, it is possible to calculate that a design for a particular building will survive the highest winds that are likely to occur in its projected life. It is this theory, and the consequences logically deducible from it, that give great confidence in structural soundness of a product, and obviate or greatly reduce the need for tests. In critical cases such as medicine and space exploration, the tests would be unacceptably expensive to conduct, or even risk serious damage to the product itself.

In my analysis, I have made several adverse remarks on the maturity of programming and programmers as an engineering profession. This criticism in no way detracts from my admiration of the enormous achievements of programmers in general, who display the greatest ingenuity, concentration, and integrity in the practice of their profession, and have made enormous contributions to the prosperity of the modern world. Programmers of today have a highly developed engineering intuition about the behaviour of computer programs, and an instinctive understanding of why they are going to work. My only criticism is that they cannot back up their intuition by appeal to scientific concepts and mathematically expressed laws. They cannot express their answers to the basic questions in terms of

mathematical formulae that could be input to the computer itself. If only they could do so, the computer itself could in principle check the consistency of a program with its specification, using standard laws of logical reasoning, as I will shortly describe.

Numerical calculation of continuously variable quantities has been extraordinarily successful in science and engineering; and the exploitation of computers has led to enormous advances in knowledge and in technology. I regard calculation as a special case of the more general concept of a mathematical proof. In the case of discrete (discontinuous) phenomena, such as computer programs, it is proof that is the most relevant technique for checking that a program is safe, and actually performs according to its specification.

Fortunately automatic checking does not have to wait until the entire program has been written. It can be applied throughout the process of engineering design, from the earliest stage, that is, as soon as the question 'What does it do?' has been answered by formulation of an adequate specification. Ideally, every subsequent design decision, detailing how the product will work, will be checked for consistency against this specification, and against all relevant earlier design decisions. And the checks can be made by mathematical proof, using as axioms a codification of the answer to the question: Why does the program work? And it is the computer itself that will give us the most certain possible assurance whether the answers to all the previous questions are correct. In the remainder of my talk, I will describe a philosophy of engineering design that has its foundation in mathematical reasoning. The basic insight is that, in the ideal, engineering design has the same structure as a mathematical proof. The basic logic and reasoning principles of rational design in engineering are the same as those of mathematical proof itself. In fact, they are the rules codified in the simplest branch of logic, namely propositional logic (or Boolean algebra, if that is more familiar).

Propositional logic

A formal mathematical proof is written out as a sequence of lines, each of which makes a mathematical statement that is always true. We will use capital letters *P, Q, R,...* to stand for the individual lines of a proof. The first line of a proof is an assumption. Each of the following lines is deduced, by application of some rule of logic, from one or more of the previous lines of the proof. The last line of the proof is often called its conclusion. The simplest rules of logic mention only the initial assumption and the final conclusion. These are separated by a conventional symbol $\vdash$ (the "turnstile"):

$$P \vdash Q$$

The meaning of this basic judgement of logic is that there exists a valid proof which begins with a line stating P, and ends with a line stating Q. Each of the (unmentioned) lines in between follows from some previous line or lines by some rule of logic. A rule of logic has a conditional form, with a horizontal bar separating a list of conditions from the conclusion:

$$\frac{\begin{array}{l} P \vdash Q \\ R \vdash S \\ \ldots\ldots\ldots \\ \ldots\ldots\ldots \end{array}}{T \vdash U}$$

The meaning of this rule is: if there exist valid proofs from P to Q, and from R to S , and from ... , then there exists a valid proof from T to U . As we shall see, the lines T and U will be expressions, usually containing copies of the lines *P, Q, R, S,...*

Our first and simplest example of such a proof rule has been known since antiquity as *modus ponens*:

$$\frac{\begin{array}{l} P \vdash Q \\ Q \vdash R \end{array}}{P \vdash R}$$

In a proof that uses this rule, the complete proof of *R* from *P* consists of two sub-proofs: the first of them proves *Q* from *P* and the second of them proves *R* from *Q*. The line *Q* occurs in the middle, and is shared by the two sub-proofs. It is the conclusion of the first and the assumption of the second sub-proof.

The more elaborate statements of mathematics can be constructed by combining simpler statements, by means of a propositional connective, such as 'and', 'or', etc. For these the standard abbreviations are:

Conjunction: V *P* & *Q* means that both *P* and *Q* are true
Disjunction: *P V Q* means that either *P* or *Q* is true, or both of them are

Special rules are provided for proving these compound propositions. The rule for conjunction is:

$$\frac{P \vdash Q \qquad P \vdash R}{P \vdash Q \,\&\, R}$$

In words: if you want to prove *(Q & R) from P*, you have to prove two things: *Q* must be proved from *P*, and also *R* must be proved from *P*. The rule for disjunction is a mirror image of the conjunction rule:

$$\frac{P \vdash R \qquad Q \vdash R}{P \,V\, Q \vdash R}$$

In order to prove *R* from *(P V Q)*, you must prove it both from *P* and prove it again from *Q*.

Propositional logic also provides a way of asserting the falsity of any proposition, but we choose to omit treatment of any such facility for negation.

Using *&* and *V* (and possibly other logical operators not introduced here), it is possible to build up a complex proposition (say, *R V ((P V Q) & X) V (S & X)*), which we will denote by *F(X)*. The *(X)* written after the *F* indicates that the complex proposition contains an occurrence (perhaps several) of the simple proposition variable *X*. As a result, the truth or falsity of *F(X)* will in general depend on (be a function of) the truth or falsity of *X*. Now every occurrence of the variable *X* in *F(X)* may be replaced (substituted) by some more complex proposition, say *(P & Y)*. In this case the result of the substitution is written simply *F(P & Y)*. A substitution of *X* by a different proposition, say *Q*, is written *F(Q)*.

Now suppose we want to prove *F(Q)* from *F(P)*. Under certain circumstances (eg. the avoidance of negation in the formula *F*), this may be done simply by exploiting the following rule of replacement

$$\frac{P \vdash Q}{F(P) \vdash F(Q)}$$

A similar rule of double replacement applies to a line which is a function *G(X, Y)* of two arguments *X* and *Y*. Obviously, it requires two subsidiary proofs as conditions to justify the conclusion.

$$\frac{P \vdash Q \qquad R \vdash S}{G(P, R) \vdash G(Q, S)}$$

The logic of engineering design

In this final section, we will argue that the correctness of a process of rational engineering design follows the same rules of the propositional calculus as a mathematical proof. However, the individual lines in the proof are much larger than the normal statement of mathematical theorems; each of them is some engineering description of the product, either in part or as a whole. The proof itself is also much longer than most mathematical proofs: it consists of the entire collection of engineering documents recording the entire design process for the product. The engineering documents describe the product in different ways from different perspectives, for different purposes, and at different levels of detail or approximation or abstraction. The most abstract documents are the overall system specifications, answering the question 'what does it do?' in terms of the properties of the product that are of interest to its users. For specifications, we will tend to use the letters *S* and *T*. Other more detailed design documents, plans, models, blueprints, etc. summarise an answer the question 'How does it work?'. For these design documents, we will use the letters *D* and *E*. The end of the design process is a detailed description of exactly how the product is to be made. For this we will use the letters *P* and *Q*. If the product is computer software, these will stand for the text of the program itself, which describes in excruciating detail exactly what the computer will do when executing the program. Although I use different letters for specifications, designs and programs, these distinctions are irrelevant to the logic.

The important relation $\vdash$ is now interpreted as stating correctness of one engineering document with respect to another. The meaning is entirely independent of the level of abstraction or detail of the documents. Thus $P \vdash S$ *asserts* that product *P* meets the specification *S*. Similarly, $P \vdash D$ means that the product *P* conforms to the design *D*, and $D \vdash S$ means that the specification *S* is satisfied by the design *D*. We even give a meaning to $P \vdash Q$, when *P* and *Q* are both either products or components: it means that *P* is a more precise description than *Q*, and therefore describes a more useful product. For example, *P* is more accurate than *Q*, in that it imposes tighter margins of permitted variation in the behaviour of the physical product. Perhaps *P* is more robust, in that it permits a wider range of variation in the environment to which it is connected. Thus any product which is known to behave as described by *P* will always behave as described by *Q* as well. But in all circumstances, *P* will be better, or at least as good as *Q*, in the sense that it is easier to predict and control - or at least as easy. Our uniform definition of the notion of correctness means that a design aid to support the process of engineering design is essentially a computer system that can construct and check proofs of logical implications.

The rule of *modus ponens* is now an exact description and justification of the normal process of stepwise design in engineering:

$$\frac{D \vdash S \qquad P \vdash D}{P \vdash S}$$

This rule may be used repeatedly, to justify a longer series of design stages separating specification from the physical implementation of the product. Each stage makes more specific choices and fills in more detail than the previous stage. At each stage, a proof is required of its correctness with respect to the design produced at an earlier stage.

A specification is most commonly expressed as a list of the requirements *S, T,* ... of the user of the eventual product. These requirements must all be met; so they must be connected by the conjunction & . Fortunately, a proof that a design D meets all the requirements together can be greatly simplified by proving that it meets each requirement separately. This separation of concerns is justified simply as an application (or repeated application) of the rule of conjunction:

$$\frac{D \vdash S \qquad D \vdash T}{D \vdash S \,\&\, T}$$

Sometimes it is desired to postpone certain design decisions until some later stage in the project, when more is known about the costs and constraints of use. The design *D V E* represents a decision not to choose immediately between *D* and *E*, but rather to keep both the options open to a later stage in the stepwise process. This postponement of decision can be justified by the rule of disjunction. This places upon the designer the obligation to prove that both the alternatives *D* and *E* will be correct.

$$\frac{D \vdash R \qquad E \vdash R}{D \vee E \vdash R}$$

A complex proposition of the form *F(X)* may be used to describe an engineering assembly which has a space allocated to hold a component *X*, but it is not yet decided what the actual component will be. If *Q* is an actual component, the proposition *F(Q)* describes an assembly in which the component *Q* has been connected in the place(s) marked by *X*. The rule of replacement states that the replacement of *Q* in an assembly *F(Q)* by a better component *P* can only make the whole assembly better. This is because *P* is more robust than *Q*, and satisfies tighter constraints, no matter what environment *F* it has been connected into.

$$\frac{P \vdash Q}{F(P) \vdash F(Q)}$$

The rule of double replacement is even more useful in the engineering design process, because it supports the strategy of step-wise decomposition. Suppose a specification can be written in the form *G(X, Y)* . Then the task of design can be split into two sub-tasks, which may be delegated to different teams working in parallel. The first sub-task is to design a component *P* according to specification *Q*. The second subtask is to design a component *R* according to specification *S*. The two components can then be slotted together into the assembly defined by the function *G*.

$$\frac{P \vdash T \qquad R \vdash S}{G(P,R) \vdash G(T,S)}$$

Conclusion

I have painted an idealistic picture of a process of engineering design, in which the correctness of each design decision in the entire design process is checked by logical proof at the very time that the decision is taken. I have suggested that these checks should be automated as far as possible in standard design automation tool-sets. These tool-sets must be based on a wide and deep understanding of the laws of the relevant branch of science. These must be formalised in sufficiently strict mathematical detail that it is always possible to calculate or prove that a product conforms to its design, and a design satisfies its specification. This can be done only in a mature branch of engineering science in which the basic foundations are sufficiently developed that the consequences of every design decision can be effectively calculated by software. This is far from trivial, since implementations, designs and specifications are usually expressed in different notations, appealing to different concepts and conceptual frameworks, and describing phenomena on widely different scales of space and time.

Nevertheless, it is my belief and hope that computer science is now approaching a level of maturity that we can envisage the construction of general-purpose and special-purpose proof tools that are strong enough to carry out the necessary calculations and checks. Their availability, in perhaps fifteen or fifty years, will revolutionise our current programming practices, and save a good proportion of today's costs of programming error.

Even when this hope is realised, the practical use of the checking tools will continue to place additional burdens on the engineer to construct specifications and other design documents to a degree of completeness and formal rigour that is needed for input to a computer check. It will always be a matter of engineering judgement to decide how far this is worthwhile on each particular project. Furthermore, the actual serviceability of the product will depend on the adequacy and accuracy of the specifications - how far they represent exactly the true requirements of the eventual user, and how accurately they formalise valid assumptions about the environment of use. No logic and no computer can ever make this essential connection between the real world and a text that purports to describe it. The availability of tools to control error will only increase the scope and value of good engineering experience, judgement, intuition, invention and common sense.

3. Plato and the Internet: Liberating Knowledge from our Heads

Dr Kieron O'Hara
University of Southampton

Kieron O'Hara is senior research fellow in Electronics and Computer Science at the University of Southampton, and a fellow of the Web Science Research Institute. He is particularly interested in the interface between technology and human society. He is the author of several books, the most recent being inequality.com: Power, Poverty and the Digital Divide (2006, with David Stevens), After Blair: David Cameron and the Conservative Tradition (2007) and Joseph Conrad Today (2007). His next book is The Spy in the Coffee Machine, about privacy in the digital age (2008, with Nigel Shadbolt). He is also the author of a book-length review 'A Framework for Web Science' with Tim Berners-Lee et al (2006), and co-editor of a special issue of the International Journal of Human-Computer Studies on 'Knowledge Representation with Ontologies' (with Christopher Brewster, 2007).

My aim in this paper is to look at corporate knowledge engineering and see what it tells us about the philosophy of knowledge. The question I am asking is whether there is anything specific in engineering that could change our understanding of what or how we know. I am interested less in generating a theory of the relationship, rather more in raising a set of questions which I hope will stimulate a dialogue between the disciplines of knowledge and engineering. The distinguished computer scientist Edsger Dijkstra, once said that "Computer science is no more about computers than astronomy is about telescopes", and in that spirit I want to argue that epistemology is no more about *people* than astronomy is about telescopes.

This paper is in five sections. To begin with, I will provide a caricature of the philosophy of knowledge. Second, I want to look at where traditional epistemology fails to connect with people's actual problems concerning knowledge. Third, I will look at the situation in reverse and think about who - or what - is the knowing subject and what epistemology would look like if actual practical problems were its starting point. Next will come a brief digression on knowledge technologies, before some tentatively-expressed (but no less firmly held) conclusions about the relationship between engineering and philosophy.

My question is a very simple one: is epistemology relevant, and if so, how? You often hear philosophy criticised as irrelevant - a criticism to which I am very sensitive, as I am trained as a philosopher. I work in the field of *knowledge engineering*, which is the discipline of understanding how knowledge is used in organisations and how to optimise its use. I have always found it very striking that there is a good deal of input from many disciplines - sociology, management science, computer science, economics, psychology, ethnography - but there is remarkably little interaction between knowledge engineering and epistemology. I find it quite strange that philosophers of knowledge are not involved in the field. How can we build bridges between knowledge engineering and epistemology, and how can we apply the rich experience that philosophers of knowledge have built up over the last 2,500 years since Plato?

People often say that they can get on without philosophy but I am not sure, in the case of knowledge, that that is true. Knowledge is incredibly important to us. For instance, we are told a great deal about the *knowledge economy* - a phrase that has become a little tatty around the edges and which we have been hearing for a long time, but which is still relevant. Knowledge, and the production of knowledge, are very important in our economy. The Lisbon agenda is still an EU strategic goal to become the most competitive and dynamic knowledge-based economy in the world by 2010 (however slowly we are approaching that goal - for a report looking at the ways that European countries use their human capital in the knowledge economy see Ederer [1]). Knowledge is a vital intangible asset in major organisations. Fixed asset values of blue chip companies only make up a relatively small percentage of their book value, but intangible assets such as knowledge provide a huge amount of the value, as expressed in market terms (Lev [2]). On top of that, of course, knowledge has always been a source of competitive advantage.

There is a clear need for knowledge management. There are some serious questions that organisations have to ask, such as how do we know what we know? If I need knowledge to perform a task, how do I know where, within my organisation (an organisation which might well exist across several nations, several industrial sectors, and may well scatter its expertise across several corporate functions), to get hold of the bits of knowledge that I need? Who knows what we need? It may be that I know that I have a problem, and I know that someone can solve it, but I just do not know who. How do I find the person who can solve my problem for me? How can we get all and only what we need, when we need it? Typically, you want that one particular piece of knowledge, but it may be concealed within a large document or repository. We spend a great deal of time trying to hack out the useful bits from the not terribly useful bits, which gives rise to the phenomenon called 'information overload'.

Let us also consider Donald Rumsfeld's incomplete matrix. Rumsfeld famously came under fire from the media for talking at the beginning of the Iraq war about known knowns, things that we know that we know, things that we know that we do not know, which are the known unknowns, and things that we do not know that we do not know, which are unknown unknowns. He became a figure of fun for saying this but he was actually talking very good sense. But if we put Rumsfeld's wisdom in matrix form, we see an immediate lacuna.

Metaknowledge\Knowledge	**Knowns**	**Unknowns**
Known	Known knowns	Known unknowns
Unknown	??	Unknown unknowns

He implicitly described the matrix but left one cell clear - the *unknown knowns*. There are things that we know, but we just do not know that we know them. That happens more often than you would think.

An example from recent history involves a company called Royal Doulton. In 2001, Royal Doulton - a big and very famous pottery company - flirted with bankruptcy and very nearly went bust. This was because it had put such a lot of emphasis on the production of large prestige dinner services, including dinner plates, side plates, pudding bowls, soup bowls and cups and saucers, on which a family would typically have their multi-course dinner around the dining table on a Sunday. However, no one was buying these because most people in the UK eat relatively casually, on non-matching plates, which they purchase one at a time from supermarkets.

Royal Doulton were employing about 3,000 people at the time, so a collapse of the firm would have been a severe blow to the economy in its home base of Stoke-on-Trent. But of those 3,000 people, most would at least have been aware of the trend towards less formal dining. The interesting point was that all the knowledge about the poor investment decisions that Royal Doulton were making was actually present within the company, but it never fed into the decision-making. That is a clear example of an unknown known - the fourth cell of the Rumsfeld matrix (cf. O'Hara [3]), and a clear example of the need for knowledge engineering.

Some epistemological truisms

Within the discipline of philosophy, one of its most distinguished intellectual traditions is the philosophy of knowledge, epistemology, usually conceived as part of the study of humanity, an important link between the philosophy of mind and our understanding of the world. *The epistemological agent is human*. We know things, and the desk does not. That is a reasonable intuition about knowledge. Other intuitions that have driven the development of traditional epistemology include:

Knowledge connects minds and the world. If I have a piece of knowledge that is in my head in some sense, as something I know, it also says something about the world outside. My mind is connected with the world via my knowledge of the world. The piece of knowledge is true: I know that Chelsea and Manchester United drew 1:1 at the weekend, because it is true. I cannot know that Chelsea beat Manchester United at the weekend, because they did not. If I know P, then P must be true. P follows from the claim that P is known.

Knowledge is propositional: knowledge says something. It can be written down as a sentence. I don't know that π, or that Manchester United, or that Brian is taller than. I do, or can, know that $\pi \approx 3.14159$, or that Manchester United drew with Chelsea, or that Brian is taller than Davina.

If I know something, I believe it. I know that Chelsea and Manchester United drew at the weekend and I also believe it. It would be very hard to imagine that I did not believe that Chelsea had drawn with Manchester United, but that I still knew it.

If I know something, then I have something beyond blind faith underpinning it. I need some kind of reason, which distinguishes it as knowledge from the rest of my undersupported beliefs. I may be able to exhibit a proof, or to point to the state of affairs that makes the known proposition true, or it may be merely that I have access to expertise that will confirm the proposition as knowledge.

Knowledge stands to belief as action stands to desire. If I desire something and the world is not like that, then I am incomplete in some way, and I will therefore want to perform some action to make the world as I desire it. Knowledge and belief have a similar sort of relationship - if I believe something without knowing that the world is like that, then I have to perform some kind of investigation. My unaugmented mind supplies an incomplete view of the world.

Knowledge undermines scepticism. If someone comes along and says, 'I think I am dreaming now. I am dreaming this room, your presence in it, even dreaming my utterance of these very words.', then I should have some kind of reply which will be supplied from the sub-discipline of epistemology. This kind of sceptical view that I may be dreaming all of this is somehow wrong, and you can say something about it.

Those truisms have come together in a view of knowledge as knowledge as *justified true belief* (JTB).

The first major epistemologist was Plato who lived in the fourth and fifth centuries BC, and whose key epistemological dialogues were the *Meno*, where he contrasted knowledge with belief, and the *Theaetetus*, in which he spoke about knowledge in terms of being a true belief plus a *logos*, where a *logos* is a kind of argument or justification or metadata. A.N. Whitehead said that all Western philosophy is a footnote to Plato, and in the case of epistemology, that is certainly true.

Knowledge is still conceptualised largely as justified true belief. If I know something then it follows that is true, that I believe that it is true and that I have some reason for believing it. Jointly, the three conditions ought to be necessary and sufficient. So knowledge is JTB.

That is not the whole story. I will not go into any kind of details but there are a great many counter-examples to this. In particular Edmund Gettier found many examples where people had justified true beliefs but no knowledge [4]. Hence, it cannot be said that knowledge *is* justified true belief, but the assumption underlying most epistemological work is still that knowledge is nearly but not quite justified true belief. So it is justified true belief, plus or minus some condition X - and the problem for epistemology is defining the X. What are the conditions that we have to put on justified true belief?

Just to be fair, not everyone thinks that and there are some dissenters (e.g. Quine [5]; Dretske [6]; Williamson [7]). Not least, there was Plato himself, who raised the question in *Theaetetus* that knowledge might be true belief plus a *logos*, but then rejected that account later on. In fact, Plato never came to any firm conclusions about knowledge.

Philosophers have persisted with this account because, since Plato, epistemology has been a reaction to *scepticism*. In Plato's day, the sceptics were fairly mild, called *sophists*. The sophists tried to explain the universe in the way it appeared and not in the way it really was. They thought, or claimed to think, that winning arguments is more important than being right. At their worst they were pure rhetoricians, but at their best they challenged a number of assumptions of the Ancient Greek elite. They were very anti-theory. However, Plato won the argument, which is why 'sophistry' is today a term of abuse, while sophistication, though something to which we all aspire, is essentially shallow.

The next wave of opponents of traditional epistemology brought us the classical sceptics, starting with Pyrrho, and including Cicero, and Sextus Empiricus. They questioned the grounds of belief. If you claim to have beliefs, the classical sceptics would question the grounds. They would for ever be worrying about why you asserted what you did. Epistemologists battled that variety of scepticism, and scepticism regrouped - the history of epistemology is a constant arms race with the sceptics. The net result was modern scepticism, of whom the classic example is René Descartes, a 17th century philosopher, who was not himself a sceptic but who invented the idea of an evil demon who could make it so that everything you believed was false.

A modern-day equivalent of this idea, and a little more scientific-sounding, is the idea that you might actually be a brain in a vat. You might think you are sitting here, reading this paper, whereas in fact you are in a laboratory, having been kidnapped by an evil scientist, your brain has been removed and he is sending signals into your brain that ape apparently sensory experience. How would that be any different for you, from your point of view? That is the classic sceptical question - that you could be entirely wrong about absolutely everything. You would know nothing about your surroundings - even if some beliefs were, contingently, true (for example that you were 93,000,000 miles from the Sun), as your justifications for them (relying on the spurious sense data generated by the scientist) were manifestly inadequate, they could not be *justified* true beliefs, and therefore not knowledge.

Epistemology is a foundational subject. It attempts to find the bases upon which we can build an edifice of knowledge, be it science or engineering. Over the 2500 years of epistemology, there have been major feedback loops so that, as epistemology beefs up with stronger and stronger arguments, so the sceptic becomes more and more radical. And so on. That is where traditional epistemology stands.

The real enemy?

I do not wish to criticise philosophy or philosophers at all for looking at the sceptical problem, because it is a very interesting intellectual problem. However, as David Hume pointed out, it is not a problem that many of us are seriously troubled by in daily life. No-one seriously thinks they are a brain in a vat. They might explore the issues in a thought experiment, but they would be carted off if they really thought they were a brain in a vat. This is a point made by many people - a common sense reaction to this, that it is all nonsense. Wittgenstein, in *On Certainty*, the last thing he wrote, argued that actually someone who comes up to you and says, 'I think I am dreaming now', or 'I think I am a brain in a vat', simply does not understand what he is talking about and this is not a real problem (Wittgenstein [8]). I am sympathetic to this.

But epistemology is not a waste of time. People do have genuinely epistemological problems, but they are somewhat different from the traditional problems of scepticism.

- Someone who lacks access to the knowledge they need. They can characterise that knowledge, but do not know how or where to find or discover it.
- Someone with huge amounts of noisy data from which they need to extract very weak signals from. Somewhere within the terabytes of data emanating from the Large Hadron Collider the Higgs Boson lurks.
- Thanks to the operation of Moore's Law, there is information overload. We can store any information we like, in huge quantities. The amount of information created by humanity per year is equivalent to every man, woman, child and baby on the planet writing a novel every single day.
- The hoarding of knowledge - we all know that knowledge sharing is useful but many mechanisms of competitive advantage mean that the incentives are more to hoard knowledge and to keep it to oneself than to share it.
- There is difficulty in reading and understanding formats and formalisms. This is a serious problem in the developing world. Information such as a weather forecast can be extremely important for fishermen and farmers, but it often comes to those fishermen and farmers in the wrong format. It reaches them in ways they just do not understand. In the more high-tech world, we have all had that grim moment where we put our disk in our machine and it says 'format not recognised', which is very frustrating. It is a serious problem when one has access to a disc that contains the knowledge they want, but the disc or data format is out of date and unsupported.
- Maintaining knowledge over time. A knowledge base evolves over time. We want to find knowledge that is out of date and remove it, without threatening completeness. We want to add knowledge to it, without threatening consistency. If the consistency is threatened, then we need to take out some knowledge from the knowledge base that retains completeness, keeps consistency and does not wreck the whole thing.
- Knowledge needs to be reliable, to be trustworthy, and trusted. We need to know its provenance. Where does it come from? Who created it? Who published it? Can I be indemnified if it is unreliable? These, of course, are particularly serious issues on the World Wide Web.

I want now to look at some types of knowledge that are hard to align closely with the JTB model and argue that perhaps traditional epistemology is rather too focused on the justified true belief model and is missing a trick, not being as widely applicable as perhaps it ought to be. I am certainly not going to suggest that traditional epistemologists have not encountered these types of knowledge, nor that there do not exist philosophical analyses that reduce them to a variant of JTB. My point is rather that a complex philosophical analysis is out of place in the context of the need for practical guidance in a real-life situation, and the requirement is for simple bridging functions that work in such contexts.

For one example, I know that the population of the world is 6 billion. It is not particularly useful to me but, if I were a demographer, or a member of the United Nations, it would be essential. It is a useful piece of knowledge, but do I believe it, and indeed is it true? In fact, I do not believe it, because I am pretty sure it is not true. The idea that the population of the world is 6 billion is almost certainly very, very unlikely to be the case - 6 billion is a round number that is a near enough approximation, not the true figure (which is unknown). There is a crucial gap between knowledge and belief. There is knowledge that we can agree on and use and exploit and deploy but I do not have to believe it.

Another example is 'know how' - I know how to swim, but I do not *believe* how to swim. In fact, I am not sure that I have any beliefs associated with that know-how. 'Knowledge how', or process knowledge, and 'knowledge that', declarative

knowledge, is a key distinction. Philosopher Gilbert Ryle is often cited with respect to this distinction, but there is precious little philosophical work relevant here (Ryle [9]). In engineering and within corporations and organisations that have goals, process knowledge is usually more important - particularly for generic accounts of practices and processes.

A related distinction is that between explicit and tacit knowledge. Tacit knowledge is knowledge that it is very hard - maybe even impossible - to characterise or write down. If I pick a glass up, I know where the glass is, but I would not be able to express what I know. I know how my body is positioned, but I can't describe it. There is a great deal of work in knowledge management, for instance a classic work by Nonaka and Takeuchi called *The Knowledge Creating Company* (Nonaka & Takeuchi [10]), about the importance within a company of getting knowledge from being explicit, so that everyone can read it in manuals, to making it implicit, tacit and internalised, and then occasionally it has to be externalised all over again. This is a very complex cyclical process.

Simply talking about knowledge in terms of belief is a relatively crude way of looking at that phenomenon. There are other related types of knowledge, such as heuristic knowledge, rules of thumb, which - literally speaking - are not true. Default knowledge is another kind, which is very important in computer science. 'All birds fly' is an incredibly useful piece of knowledge, although it is of course not true because penguins do not fly. Nevertheless it is very useful, if you are told that Tweetie is a bird, to assume that Tweetie flies. We have default logic, but it is not clear that we understand the semantics of default logics very well. To say that truth is involved here is rather a distortion.

Then there are bodies of knowledge. How much knowledge do we store? How much do we outsource? By outsourcing, I mean that we might make private notes, computer files, books, or we might know that it is on the Web. There is a question about whether we need to worry in principle about knowledge stored in the head. Is knowledge in the head really a special case? It goes in, and then it may or may not go out again. In the *Phaedrus*, Plato reports Socrates as worrying that all our brains will become flabby if we write everything down. Plato was writing at the beginning of Greece moving from an oral society to a literate society and he was particularly worried that we were putting our mental abilities at risk.

The answer to this - to Socrates' worries about writing things down, is - well, so what? It is not too much of a worry for me. Why focus on the psychological state? You may just want to convert 'knowledge that' into 'knowledge how', so you learn something and you write it down - and now I know how to get at it, because I know where I wrote it down, or I know the URI of the webpage where I made a note of it. So I can get at the knowledge if I need it and I do not bother to store it. Indeed, there are good arguments that reconfiguration of the environment to support cognition is an important aspect of our human condition (Clark [11]).

Automation is another key area where simply viewing knowledge as belief is misleading. Consider expert systems. Routine procedures encode a huge amount of knowledge in corporate memory, and data mining. There are many areas where no one person knows the procedures; they are all performed by computers, or systems, or they are distributed over large numbers of people. No one knows the input data, because they are sucked down from the Web or the output of some data-producing computer. Inference is automated, and output feeds directly into management. There could be a huge set of knowledge-based procedures going on, where input is converted to output and no human being actually appears in the loop at all - that happens reasonably frequently nowadays, now that we have the technology to do that. Looking at knowledge as a species of belief, *simpliciter*, is a bit tricky. Epistemology is certainly relevant, but the philosophy of mind less so.

We have to conclude that the traditional epistemological project that Plato envisaged has actually not succeeded terribly well. We do not have agreed foundations for knowledge and the sceptic is still a worry for philosophers. We may say that knowledge is justified true belief, plus or minus X, but we do not know what the X is, and we have failed to solve Plato's problem. This is partly because the problem keeps evolving every time we get a new solution to it. Failure to solve a problem is not a proof of the incorrectness of the JTB theory, but it is a strong indicator. It certainly does not mean that traditional epistemology has provided no insights, but its insights are coming to us - as knowledge engineers at least, and organisational managers - in the wrong form.

The knowing subject and the general epistemological problem

Given all this, do we want to cling to the view that the knowing subject is always human? Once you insist on knowledge being a species of belief, then it is a very quick route to saying, in that case, that all things that know are human (or maybe animal, a deep discussion I certainly don't want to go into). But aren't there other candidates for knowing subjects? A case could be made about a number of agents having at least some of the attributes that we would want to apply to knowing subjects.

First of all, there are several artificial devices about which it is not *prima facie* absurd to say that they know things. We say that at a very high, shorthand level all the time. We invent computer game characters or other avatars that have epistemological problems to solve - they need information and they need knowledge. They investigate independently, and they draw conclusions from data.

Secondly, systems use information, churn it up and push it out, as do organisations and companies. It seems to be perfectly reasonable to say, with the anthropologist Mary Douglas, that institutions think, that organisations are knowing subjects (Douglas [12]). They have a number of characteristics common to uncontroversially psychological agents.

Thirdly there is distributed knowledge, knowledge that is not focused or stored in any particular place but it is spread about the place. Hayek talks about the signal given by a price, a heuristic measure of distributed knowledge about demand and supply determined by a competitive bidding market (Hayek [13]). No one person knows all the facts about who wants to buy and sell some particular commodity; all one needs to know, to take part in a market, is whether one is prepared to buy or sell at any given price - an extremely low epistemological overhead. But when all the deals on a market are put together, in the context of a very large number of transactions, we get a price that is a heuristic measure of scarcity and demand. There is knowledge about goods in a market, but it is distributed across the buyers and sellers, and the price is a heuristic aggregator.

Fourthly, consider Google and its like. Who would have thought that taking the eigenvectors of the link matrices of the worldwide web would actually give us anything sensible about what conversations were going on, and what key words were of interest to other people? Google has become a giant company on the back of that scientific insight and the engineering skill to enable the PageRank algorithm to work at the Web scale, in an environment much more hostile than most typical information retrieval contexts (Batelle [14]; Page et al [15]).

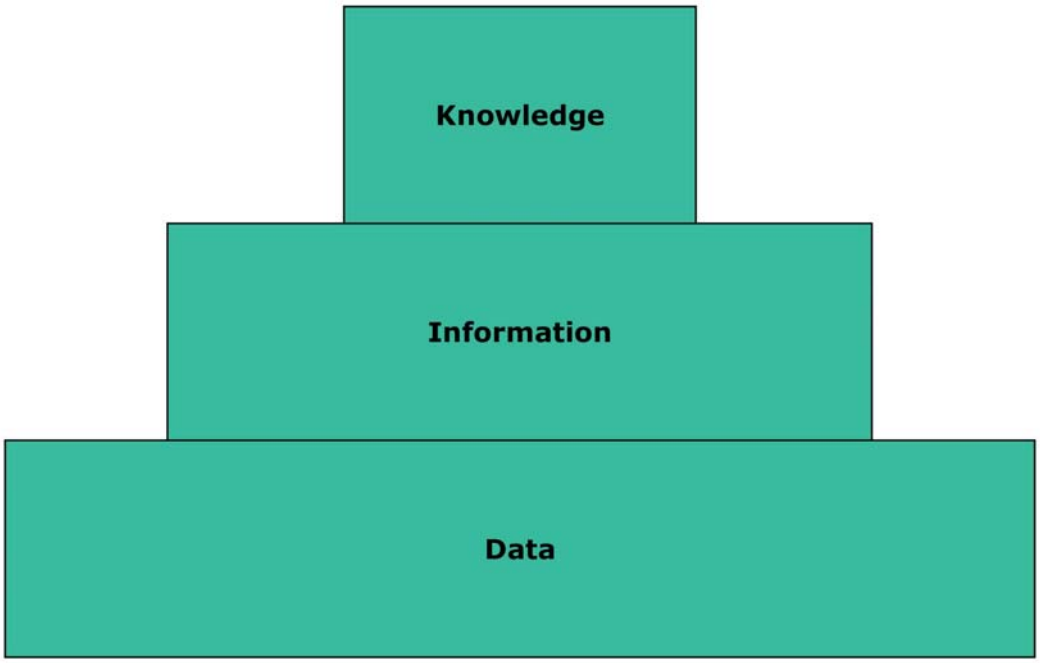
In the computer world we also have the example of Web 2.0. 'Folksonomies' are emergent conceptual structures that develop from lots of people making individual little descriptions of something like a photographic storage space, like Flickr. Interesting structure comes out of that aggregation created by no-one.

Knowledge is not the prerogative of human beings. Knowledge is a kind of interaction between an agent and the word, which began, in primitive form, with unicellular creatures taking material from outside its own outer membrane and excreting material back. It is "learning" things in a very crude way about its environment and putting things back into the environment to change it. Autonomous living systems, when they separate themselves off from the rest of the environment, are free to decide, in a sense, what they will "notice" in the outside world, and what will disturb them. This is a feedback mechanism: systems can alter the environment by pumping out waste products and signals in various ways (Capra [16]). This principle of engagement with the environment is a primitive epistemological mechanism - a trick that humans have learned, as we restructure and reconfigure our environment in order to lower our epistemological and cognitive overheads (Clark [11]). Put another way - and this is a key point in particular for engineers - the world is not exogenous. Epistemology is not simply the passive mapping of a world whose properties are outside the system. Most epistemological projects worth their salt bring the world into the system, make it endogenous. We can, and do, change it as part of our investigations. Engineers, of course, are specifically mandated to change it (a point made following my presentation of this paper at the Royal Academy of Engineering by Alan Morris of Cranfield University).

This leads me to characterise what I call the general *epistemological problem*. There is an agent in an environment. That environment could be the real world, or a virtual world, or the Internet, or a specific website or any space where things can act. It could also be some highly circumscribed world, like the world of business, or a competitive bidding market, that might be described with a relatively circumscribed set of parameters.

To operate in the environment, the agent needs information. If it wants to act, it needs to defend and preserve itself. It will have goals to achieve (possibly including reproduction), it will want to adapt to the environment, and it will want to change the environment and adapt it to itself. All of this demands information about the environment, extracted dynamically, possibly opportunistically, and the evaluation of that information over time. The information needs to be reliable. The basic epistemological question is how, given that, does the agent extract *usable* information from that environment? That seems to me to be the general form of the epistemological problem, but of course it applies not only to people but also to animals, companies, organisations and artificial agents as well. The problem to be overcome is not scepticism and it is not that you think that you have knowledge but really you do not - that is not the problem. The problem is failure. You need survival, not foundations. That is an engineering way of putting it - we want not to fail, or to fail safe.

In this context, I would want to argue that knowledge is not best understood as justified true belief, or indeed as related to justified true belief in any strong way. You can imagine knowledge on top of a pyramid. Knowledge is basically usable information and it is a type of data. What is information? It is meaningful data. We can go down until we meet raw symbols, but the point, for epistemology, is that we need to be reasonably sure that the information we extract from the environment will facilitate the goal-seeking behaviour of the agent in the appropriate timescales, given the agent's information-seeking apparatus, and given its processing power. If it takes in more information than it needs, it needs to be able to separate the important information from the less important. It does not need to be perfect (the scepticism question), and "discovering" something false every so often is not a serious problem as long as the information it acquires can be used to promote its interests often enough for the agent's purposes.



The data-information- knowledge pyramid

When Plato wrote the *Meno* and the *Theaetetus* - great works both - humans and animals were the obvious knowing subjects and it was not really clear that anything else could plausibly be such a subject. However, the characterisation of epistemology above subtends a continuum: there are non-human species, there are individual humans in everyday life (the paradigm case of things that know), and there are also groups or systems of humans in specialised contexts. One such context is science, which is a particular way of learning, and engineering is another (one that has not been written about very much). Other contexts include art, and free markets - in all of these contexts groups of people, not necessarily well-co-ordinated, seek knowledge and information in particular ways. Then there are more structured organisations, where the obvious examples are companies and states. Then there are technologies, which I will consider in a little more detail.

Knowledge technologies

There are many technologies for engineering knowledge within organisations, so that organisations can find the knowledge that they own successfully. The World Wide Web is the biggest and most complex technology ever created, a giant information store of linked documents. We have indexing by direct association and it is completely democratic. Any page can link to any other page and there is no hierarchical structure. The result of that basic vision is that amazing information store that is the World Wide Web. Berners-Lee has now moved on to trying to develop what is called the 'Semantic Web' which extends the Web by allowing direct linking and manipulation of raw data. Whereas the Web links documents, the Semantic Web is supposed to link actual data. If it takes off, that will be much more powerful (Berners-Lee et al [17]).

Then there are ontologies, which express shared conceptual schemes. These are incredibly useful in allowing human and artificial systems to talk to each other. That co-operative problem is a serious epistemological issue. Let us not forget search, which is massive: that is why Google is one of the largest companies in the world (Battelle [14]). Search is the problem of our times. We are learning to say more things *about* the knowledge that we have and so we not only have knowledge but we have knowledge about our knowledge - metaknowledge, annotations, metadata. Then there are technologies for user profiling and personalisation, which are useful in knowledge-based contexts. The technology actually knows a little bit about you and in effect changes the virtual environment in ways to lower your epistemological overhead.

One example of a specific knowledge technology is the Conceptual Open Hypermedia Services Environment (COHSE - Carr et al [18]). In Web documents, the author of the page has put in the links, so you get his or her associations. The links you can click on in a document are determined by somebody else. The idea behind COHSE is a very smart one. Because the system knows a little about you and your interests, and is able to search the web, it actually provides links that have not been provided by the author of the paper or the document, and instead are ones that you might be interested in. So, in a very real sense, your past browsing behaviour - which is what is used to create the user model - helps to point you towards new Web pages and concepts that you would not otherwise have thought of. That is specific to you, and it is a nice idea about the way that we can use knowledge and technologies together to make our knowledge acquisition more efficient. A major multi-million pound project funded by the EPSRC upon which I worked for some years, Advanced Knowledge Technologies (AKT - www.aktors.org, and see Shadbolt [19], Shadbolt & O'Hara [20], Shadbolt & Kalfoglou [21], Shadbolt et al [22]), was devoted to exploring the possibilities for the use of knowledge technologies to support agents' and organisations' knowledge acquisition and curation.

Conclusion: the role of philosophy

To conclude, it is important to clarify the role of philosophy with respect to its object disciplines, including engineering. What can the philosopher say to affect engineering practice? Many engineers are, I believe, overly-sceptical about what the discipline of philosophy can contribute. I do not think it is the role of the philosopher to say everything that there is to be said about ideas. Ideas change - ideas have histories, concepts have histories - and technological development can and does influence that change. There are many interesting examples of ways that technical change has influenced conceptual change. For instance, our ideas of the mind have altered in reaction to discoveries in medicine and the neurosciences.

We need to think about how philosophy and science and engineering can interact to produce mutually beneficial outcomes. It is always important to realise that any human action is improved by the existence of a reflective metalevel. It is always useful and interesting and important to think about what you are doing, in relation to your other life-projects, your (social and physical) environments. Abstract characterisations of engineering, such as we have seen debated at The Royal Academy of Engineering, generate a lot of argument - which is an indication of how important engineers believe them to be. The reflective metalevel is an intellectual construct which philosophers are of course highly capable of populating and debating.

Scientific method is a good example of a philosophical metalevel that performs real work. Birds sometimes need ornithologists to stop them dying out and the same may well be true for scientists. With constructs like scientific method, we have a pretty subtle understanding of science, how it operates epistemologically and as a social activity. The major thinkers in scientific method, from Popper and Kuhn back to Francis Bacon, have made an incalculable contribution to the self-conscious pursuit of science. Philosophers have not been the only contributors and scientists themselves have of course improved scientific method as well. However, when we look at new areas of science, such as e-science, it is quite clear that philosophical analysis has made its contribution. And methodological questions are beginning to loom large in engineering - witness The Royal Academy of Engineering's Philosophy of Engineering initiative. The moment that you start thinking about engineering as an abstract thing and about methods and methodology, philosophers can have a very important role to play.

And getting back to knowledge and epistemology, think of philosophers' 2500 years of experience. The focus on justified true belief and the human as the knowing subject is not irrelevant, because most knowing subjects are human and most knowledge is probably reasonably characterised as justified true belief. It is a matter of great frustration to me that that experience is not (often for presentational reasons) imported into my field of knowledge engineering very often.

My take-home message is that science, social science and philosophy all have a role to play in a joint project to understand how usable information is to be gathered from environments, by agents human and non-human, natural and artificial, and how artificial environments can be structured to maximise the amount of usable information that they provide. That is how we should all be fitting together, and that is the project or higher goal to which we should all raise our eyes.

References

1. Ederer, Peer (2007) *Innovation at Work: The European Human Capital Index*, Lisbon Council Policy Brief, **www.lisboncouncil.net/media/lisbon_council_european_human_capital_index.pdf**
2. Lev, Baruch (2001) *Intangibles: Management, Measurement and Reporting*, Washington: Brookings Institution
3. Kieron O'Hara (2002) *Plato and the Internet*, Duxbury: Icon Books
4. Edmund Gettier (1963) 'Is justified true belief knowledge?' *Analysis*, 23, 121-123
5. Quine W.v.O. (1969) 'Epistemology naturalized' in *Ontological Relativity and Other Essays*, New York: Columbia University Press, 69-90
6. Dretske Fred I. (1981) *Knowledge and the Flow of Information*, Oxford: Blackwell
7. Timothy Williamson (2000) *Knowledge and its Limits*, Oxford: Clarendon Press
8. Ludwig Wittgenstein (1969). *On Certainty,* Oxford: Blackwell.
9. Gilbert Ryle (1949) *The Concept of Mind*, Harmondsworth: Penguin
10. Ikujiro Nonaka & Hirotaka Takeuchi (1995) *The Knowledge-Creating Company: How Japanese Companies Create the Dynamics of Innovation*, Oxford: Oxford University Press
11. Andy Clark (1997) *Being There: Putting Brain, Body and World Together Again*, Cambridge, MA: MIT Press
12. Mary Douglas (1987) *How Institutions Think*, London: Routledge
13. F.A. von Hayek (1945) 'The use of knowledge in society', *American Economic Review*, 35, 519-530
14. Battelle, John (2005).*The Search: How Google and its Rivals Rewrote the Rules of Business and Transformed Our Culture*, Boston: Nicholas Brearley Publishing
15. Page, L.; Brin, S.; Motwani, R. & Winograd, T. (1999) *The PageRank Citation Ranking: Bringing Order to the Web*, Dept. of Computer Science, Stanford University, Technical Report 1999-6
16. Capra, Fritjof (2003) *The Hidden Connections: A Science for Sustainable Living*, London: Flamingo
17. Berners-Lee, Tim; Hall, Wendy; Hendler, James A.; O'Hara, Kieron; Shadbolt, Nigel & Weitzner, Daniel J. (2006) A framework for Web Science', *Foundations and Trends in Web Science* 1(1), 1-134
18. Carr, Les; Bechhofer, Sean; Goble, Carole & Hall, Wendy (2001) 'Conceptual linking: ontology-based open hypermedia', *Proceedings of 10th World Wide Web Conference*, **www10.org/cdrom/papers/frame.html**
19. Shadbolt, Nigel (ed.) (2003) *Advanced Knowledge Technologies Selected Papers*, Southampton: University of Southampton, **www.aktors.org/publications**
20. Shadbolt, Nigel & O'Hara, Kieron (eds.) (2004) *Advanced Knowledge Technologies Selected Papers 2004*, Southampton: University of Southampton, **www.aktors.org/publications**
21. Shadbolt, Nigel & Kalfoglou, Yannis (eds.) (2006) *Advanced Knowledge Technologies Selected Papers 2005*, Southampton: University of Southampton, **www.aktors.org/publications**
22. Shadbolt, Nigel; O'Hara, Kieron; Heath, Tom & Tuffield, Mischa (2007) *Advanced Knowledge Technologies Selected Papers 2006-2007*, Southampton: University of Southampton, **www.aktors.org/publications**

Part II: Systems Engineering and Engineering Design

4. The Context and Nature of Engineering Design

John Turnbull FREng

John Turnbull graduated in Chemical Engineering in 1961 and joined BP. His early career was in process development and design. He was subsequently involved in production management, business management and then the management and direction of technology, engineering and HSE. After retiring from BP in 1993 as Deputy CEO of the then BP Chemicals he engaged in general management consultancy and the direction of an international senior executive programme for the Wharton Business School in the University of Pennsylvania.

The design engineer plays a pivotal role in shaping society and its lifestyle and values - particularly in this modern, technology-driven age - and yet, since the age of Brunel, Stephenson and Telford, engineers seem to have retreated and become largely anonymous background figures. This is a serious concern because, with due respect to John Donne, 'no engineer should be an island'. It goes against the nature of the discipline. Given the purpose, role and impact of the design engineer in society, surely he or she should be much more upfront, a much more public figure, and should play a much more active role in the community.

While preparing this paper I found three reports produced by The Royal Academy of Engineering which are highly relevant to the topic. These are *Educating Engineers in Design* (2005); *Common Methodologies for Risk Assessment and Management* (2003) and *The Societal Aspects of Risk* (2003). All can be found on The Royal Academy of Engineering website.

What is engineering?

Let me explain why I consider design to be so important to engineers. This definition of engineering appeared in an earlier Academy report [1]:

Engineering is the knowledge required, and the process applied, to conceive, design, make, build, operate, sustain, recycle or retire, something of significant technical content for a specified purpose; - a concept, a model, a product, a device, a process, a system, a technology.

This was intended to be an all-embracing and comprehensive definition. But its somewhat legalistic approach is, I think, misleading. It does describe what many engineers do, but it masks the really core and fundamental engineering activity which is *design*.

If we dissect the definition we can say that '*applying knowledge and skill*' apply to any profession. But the *conception and design of systems* are the fundamentals of engineering. The activities covered by '*make, build, operate, sustain*' are essentially *management* activities requiring skilled, well-trained personnel, to work according to the designer's recipes and instructions. They may or may not be engineers in practice but, in essence, it is the design that directs their work.

Many years ago I was responsible for running a petrochemical complex and the last thing I wanted was an engineer to actually run the plant. He or she would be forever looking for ways to improve the operation and to modify it, when what was really wanted was someone who meticulously followed the operating instructions as per the design. Equally, we want pilots to fly aeroplanes, not aeronautical engineers.

It is of course normal to find faults with a design after a period of operation, and I do not deny that engineering expertise is then required. But that is essentially an offline activity and a design project in its own right. It is an activity separate from the business of '*operating, sustaining*' etc. referred to in the definition.

Engineering is about *significant technology*. But perhaps the most important word in the definition is 'purpose': '*for a specific purpose*'. Those other definitions of engineering design which say '*to meet requirements*' are quite unsatisfactory. This is far too passive. Engineers have a much more active role in helping to define the optimum requirements.

Finally, the design engineer should always see his product as a *system, or a part of a system*. As discussed later, a design must have a defined context.

What is 'design'?

Dictionaries take pages to cover the meanings and nuances of the word 'design', and I am astonished that it seems to be involved in everything from matchmaking to homicide. But I like this definition from Webster, which appears to encapsulate what engineers mean by design.

The process of selecting the means and contriving the elements, steps and procedures for producing what will adequately satisfy some need.

However, given the philosophical context of this paper, I suggest that we can obtain a fascinating insight into what many people think that design is about by listening to the advocates of 'Intelligent Design' in their arguments against evolution. This insight is separate from the conclusion that is drawn. It seems that the supporters of 'Intelligent Design' see in the Darwinian concept of natural selection a kind of random, unstructured process. They fear that this robs human existence of meaning and they argue that the sheer complexity of the natural world demonstrates, or even proves, the existence of a 'designer'. Purpose, in their minds, deserves particular emphasis and seems to be missing from natural selection. Whether this implies that they see design - 'engineering design' - as a god-like activity, I do not know. However, it is clear that they see design as a very high level activity that can bring order and meaning to life. Engineers should say Amen to that!

Engineering design process

Other papers from this seminar have much more to say on the techniques of engineering design, but I would just like to emphasise the crucial nature of the upfront conceptual and outline design phases. These involve intensive two-way dialogue with the client. A close fit has to be developed between the client's business plan and risk model and the strategic elements of the proposed design. Both the designer and the client must agree on the boundaries of the intended system and this must be done, not only in terms of the topography, but also in terms of time. 'For what period of time is this system expected to operate?' is a key question.

Both parties need to share a common risk assessment and management process that takes into account the inevitable uncertainties. This is particularly important in terms of the technologies to be employed; the areas where the design will require creativity and judgment. Finally, it must include a financial model that expresses and tackles the key economic uncertainties from both the client and designer viewpoint.

In summary, this is the period when the design engineer ensures that there is a clear understanding of the purpose of the system to be delivered by the project and his or her ability to deliver it.

Like most professionals, engineers' success or failure can be measured on a scale between 100 and 0 per cent. But engineers are usually subject to a very transparent and public test. Engineering designers must produce products and processes that work. Everyone can observe, and suffer the consequences if bridges fall down, if aeroplanes do not fly, and automobiles will not start. The consequences of error are much more obvious than in the work of, say, a neurosurgeon, lawyer or accountant.

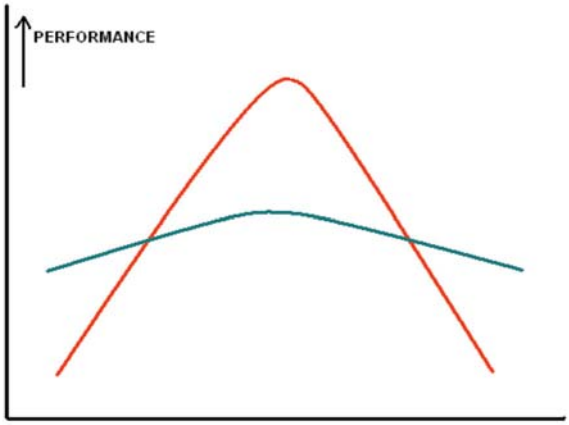
There are engineering failures, but generally engineers can and do produce complex systems that work and this is *the* very specific skill that gets to the heart of what engineering is about. This appears to have significant philosophical relevance. Much of the traditional morality, culture and behaviour that we know are based on what worked for successive generations, in terms of behaviours, family loyalty and even diet. It seems to me that philosophers, together with religious leaders, have worked more or less successfully through the ages to develop logical and coherent structures to explain these traditional values and behaviours, although they generally have a pragmatic basis. Some of these "structures", or philosophies, even invoke penalties and rewards way beyond the commonsense basis for these behaviours - some even apply after death.

But of course, society has changed, especially in the last couple of hundred years or so and it has become much more complex. People are better educated and informed and question conventions. We, as engineers, have driven much of that change. Birth control has changed family structures and behaviour. Communications technology puts us in contact with a much wider and diverse community. Even at the level of food hygiene, refrigeration has changed the rules. We surely have to revisit our traditional rules of behaviour and I would suggest that philosophers and religious leaders as well as the rest of us should be doing this, using the engineer's yardstick: *What will work best?*

Encouragingly, we increasingly hear politicians and others asking questions like, "Does the educational system work?" "Does the tax system work?" "Does the NHS work?" "Does alcohol licensing work?" "Does the traditional family work?" Like engineers, the wider world is perhaps realising that evidence-based, empirical solutions developed in a logical manner are more effective than solutions based on ideology, prejudice and emotionally invasive principles.

Another key characteristic of engineering design is the way in which it involves a large number of trade-offs and compromises. Behind most advantages there is a cost; behind most savings, there is a performance penalty. There is no ideal or perfect solution, but there are many possible solutions and the skill of the engineer lies in producing the best or optimal one. There is no such thing as a free lunch: Single issue pressure groups, please note!

One pitfall to be avoided in the selection process is to design a system that requires continuous tuning and adjustment to sustain high performance. A system whose performance peak lies in a very small section of its possible operating envelope is generally much less useful than one with perhaps a lower peak, but covering the majority of the operating range.



The petrochemical complex referred earlier was a green field design and involved 14 interconnected plants. The design was truly elegant. And if all of the plant operated at 100 per cent capacity the complex had world-beating efficiency and economics. But, of course, in the real world, every plant could not always operate at 100 per cent output. With the various plants operating at different levels of their design capacity the complex was a pig to control and certainly did not achieve the hoped for economic performance.

Engineering design process

In a world of accelerating change, flexibility is of increasing importance. When I started my career, I worked with a boss who persuaded me to join him in designing what he called a 'standardised crude oil distillation unit'. This was at a time when building refineries in Europe was very much in vogue. His idea was that, if we standardised the design, there would be considerable economies derived from a sort of 'production line' approach.

Our practice at that time was to design such units to be able to process efficiently light crude from Iran, and heavy crudes from Kuwait. We did just that in our new design and went up to Head Office where we gave a brilliant presentation on the possible operating and capital savings of our concept. The then general manager of the refineries department said, "Excellent concept! But actually, I want units that will process not only Middle East crudes, but also as yet undiscovered crude oils from places like Patagonia." We cursed under our breath, retreated back to base, and decided we did not want any further dealings with this arrogant fellow. But, of course, as things turned out he was right. Within 10 years our refineries had to process crude oils from the North Sea, from Libya, from Indonesia and wherever. The Middle East was no longer the sole supplier. We should have listened to him and revisited our design concept.

Shortly after that, I was involved in the process to convert oil into single cell protein. With crude oil at $3 a barrel, we could feed the Third World and the logistics were absolutely straightforward. However, just as we completed the plant, the price of crude oil quadrupled to $12 a barrel: End of project!

Climate change is the big topic of the moment. I used to be responsible for the world's biggest synthetic ethanol production units. We had found that making ethanol synthetically from oil was much cheaper than fermenting cane sugar, molasses and the rest. As a consequence of climate change we now have the situation where we are told that ethanol produced by fermentation should replace petroleum fuels. This is a sudden and complete reversal of the economics and the thinking of the last 60 years.

These may be three rather extreme examples, but they illustrate that we live in a very dynamic world and we must not design as if it were static.

Societal risk

The design engineer has to take account not just of technology and economics. There are significant non-financial benefits and disbenefits to take into account. Beyond the client, there are other important stakeholders, although their identity may not be immediately obvious and effort is required to establish who they might be. As engineers, we need to understand that these stakeholders will often have a completely different agenda from our client and ourselves. They may even see the boundaries of our system quite differently. We saw this in dramatic fashion when Shell designed the de-commissioning of Brent Spa. Greenpeace had a completely different agenda from that of Shell and succeeded in derailing the Shell design.

We, as engineers, like to quantify as much as possible but there is significant territory where quantification is fraught with problems and perceptions count as much as reality. As UK engineers, we are all proud of the magnificent job that BAA has done to date with Terminal 5. But I am sure, given the direction of the prevailing wind, the residents of Central London, Richmond and Windsor would like them to have followed the French example and instead worked to move London's major airport either to the north or the south of the Capital. This is a clear case of different stakeholders failing to agree on the boundaries of the system. The Academy's report on *The Societal Aspects of Risk* deals fully with the issues of stakeholder involvement in the design process, and I commend it to you.

Aesthetics and utility

When discussing "design" in the wider community, engineering design does not arise as a first thought or example. I suggest that at your next dinner party, you invite your guests to name a designer, and I wager that they will reply with names like Dior, Terence Conran and Laura Ashley. But they will not come up with an engineer.

Our bank recently wrote and offered us a replacement cash card, which they said had been "designed" by Stella McCartney. I was quite impressed to find that a fashion designer had mastered the technology of polymeric materials and their embossing and lamination, not to mention imprinting a magnetic strip, incorporating a chip and the necessary encryption technology. However, when I read on, I discovered that what the bank meant by "design" was that she had provided a pretty picture to put on the front of the card. But we cannot ignore or dismiss this interpretation, because good aesthetics are valued and respected. On the other hand "utility" is taken for granted mainly because we engineers do such a good job.

Architects combine utility and aesthetics, but can engineers; and should they? Of course they should - and sometimes, they do.

Engineering and aesthetics

Automobiles are highly engineered systems, but we well know that they are bought as much for their appearance and style as for their performance. Concorde is a dramatic example of how, even when designing for performance, something aesthetically satisfying can result. I am astonished to find, even today, how many people talk in sentimental, nostalgic terms about that beautiful, wonderful machine. It really was such a beautiful aeroplane, they say, and they appreciate that. Of course, they appreciated its performance also, but what they really emphasise is its looks.

If you do not know it already, type "Millau Viaduct" into Google and you will find lots of photographs of this most beautiful bridge. Asking some of my (better informed) friends who actually designed it, they tell me that it was designed by Norman Foster. Norman Foster, of course, made a magnificent contribution, in terms of the outline and the shape of the bridge, but the man who actually ensured that that bridge stands up and can take the traffic and resist the elements is called Michel Virlogeux. In the UK, outside of the structural engineering community, I doubt whether anyone has ever heard his name.

Then there is the London Eye: I think it is a super, beautiful object, but I have friends who think it is the greatest blot on the landscape. Wherever we look we discuss and appreciate aesthetics, but we take the engineering for granted.

You will have your own list of systems that combine good engineering and aesthetics. And I would argue that good aesthetics and engineering can and should live together. An aesthetically satisfying design will normally enhance public acceptance of the technology.

Purpose and other values

However, when reflecting on purpose, apart from utility and making it work, and apart from aesthetics, some other values begin to appear. There are ethics, for sure; a social focus, because people are concerned with health, education and care for the elderly; environmental responsibility and sustainable development; and engagement with the developing world.

Fascinatingly, of course, these are issues that are rising up the engineering agenda more and more today. The list is not complete, but these are issues which are certainly receiving the attention of the Fellows of this Academy, and an increasing number of reports and studies are being generated to address these values. I am certain that this trend will help to increase the public awareness and appreciation of engineering, because it addresses areas that count out there in society. By addressing them we are seen to be responding to society's wider agenda.

Engineering's social dimension

In conclusion I want to argue that engineering is not technology driven, nor should it be. Technology is an enabler and it is the key component in the toolkit, but it is not the reason why we are engineers: our real driver is social, because we want to improve the quality of life out in the community. However, what distinguishes engineering from other equally socially-driven professions is its range of activity, its ability to combine science with judgment and intuition; and all of this within a disciplined, technical framework. Engineers have the skill and ability to design, according to well-tested rules, complex systems that work. They can combine science-based technologies with social insight, to improve the quality of life.

Engineers need to be as adept at functioning in a wider political environment as they are in a technical one if they are to fulfil their role

So why don't engineers enter the political field? Their methodology, skills and talent are surely sorely needed there. Natasha McCarthy and I have looked to see how many professional engineers we can find in the House of Commons as Members of Parliament and I think we may have found 2.5. Of course, it is very difficult to spot them amongst all of these lawyers, teachers, doctors and economists. Why do we engineers feel so bashful about contributing to society, when we have the skills that we have?

The Academy does a great job as the voice of engineering, but is that voice diluted by the plethora of institutions representing the professions? There is a great deal of noise out there, and the message gets lost. Too often the media and the general public and, indeed, some government departments, make the mistake of lumping science and engineering together. But we need to make it clearer in the public consciousness that science is quite a different discipline from engineering. It is an objective, knowledge-seeking discipline. It is not seeking to make people either happy or sad, but it is looking for laws of nature that are as valid in Aberdeen as they are in Zanzibar.

There seems to be worry, fear and anxiety about science out there today and I do not know why. But engineering should be seen quite differently. While engineering makes use of the discoveries of science, it is not indifferent to people and their aspirations. Good engineering is attuned to the culture of the community that it seeks to serve. Good engineers know that what works in Japan will not be the same as what works in Arizona and respond accordingly.

Finally - and I hope that I am out of date when I say this - engineering formation needs to recognise and give much more room for the social skills that a professional engineer needs. To be taught, as I was 40-odd years ago, that engineering is 100 per cent technology is a travesty of the truth. The engineer's mission must surely be to serve the community and, to do this, the engineer needs to have the skills of communication and debate to engage the community and to address and educate its needs and aspirations.

References

1. Sir Robert Malpas (2000) The University of Engineering: *A UK Perspective*. The Royal Academy of Engineering

5. Philosophical Issues in the Practice of Engineering Design

Professor David Andrews FREng
Department of Mechanical Engineering
University College London

David Andrews was given a new chair in Engineering Design at University College London in 2000 following a career of over 30 years in the Ministry of Defence in naval ship and submarine design and acquisition. He was the Project Manager for the Replacement Amphibious Fleet, Head of Preliminary Design and, finally, Director Surface Ships. In two earlier academic secondments to UCL he developed a new approach to computer aided preliminary ship design. His current research team at UCL is focused on exploiting this approach across a wide range of applications, including novel ship types, design for production and integrating simulation techniques into initial design. He also edits the Design Methodology State of Art Reports to the tri-annual International Marine Design Conference.

The nature of engineering design

I would contend that it is largely the design element in engineering practice that distinguishes engineering as an activity from the sciences. Clearly, there is both science and art in engineering practice and it might be appropriate to look at the philosophy of science, as a well-developed field of study, for insights into engineering as a discipline. By which I mean both engineering as a whole and the distinct practice of engineering design. Much of the affinity of engineering to science clearly lies in the practical application of scientific insights, which were obtained initially to better explain the physical world. Then that knowledge - often called the engineering sciences - is used particularly to undertake that extra special activity that is engineering design.

Much of what an engineer does in the process of design, which distinguishes engineering design from most other design endeavours, is applying scientific analysis using scientific knowledge. But of course, in using the engineering sciences, we as engineers do not do so in a pure scientific manner. We are pragmatists, we want answers and we will use what knowledge there is to best progress our analysis - often having to make scientifically unverifiable guesses, employing scientifically informed but often scientifically unjustifiable assumptions and simplifications. These, of course, are only justified in the pragmatic sense of getting a workable solution to the immediate task in hand. So, even at the scientific end of engineering activity, our practice is inherently different to that of science, and this is even more the case with engineering design.

The other aspect which is often said to characterise engineering design is its creative nature. Of course, creativity is a very difficult issue and even artists and other non-engineering based designers are very wary of trying to pin it down, it seems, almost for fear of destroying what is often considered to be some kind of 'magical' element. Very often the creative element in design is thought to be confined, in engineering, to the concept creation or early synthesis and that, once the design's essential form has been created, in a divergent or often brainstorming manner, the rest of the design process is seen to be more rational and convergent. It is then that the engineering sciences can be fully applied to analyse the new concept for its technical feasibility and evaluate the results of that analysis to make decisions on what is acceptable and how the design should progress. Eugene Ferguson, in *Engineering and the Mind's Eye* [1], seems to counter this by pointing out that, throughout the process of designing an artifact, engineering designers actually make dozens of small decisions and hundreds of tiny ones.

Why is design different from science?

So is the design element in engineering significantly different from the science in engineering? There would seem to be two aspects to this. The first applies specifically to engineering design as a field of engineering practice and is the recognition that the practice of engineering design extends far beyond the direct application of science (by which I mean the classical engineering sciences) to those many other disciplines relevant to the use of the design product, such that the practice is multi-disciplinary.

Many design issues are associated with the human application of the designed product and are crucially important to the designer's decision making. These include factors such as economics, societal pressures and constraints, be they legal, ethical, cultural or perhaps even political, as well as ergonomic and psychological aspects. All these mean that the engineering designer has to be much more than just an applied physical scientist. He or she also has to be skilled in the human sciences and all of these topics seem to require ever greater breadth in the designer's awareness.

The second aspect, is engineering design links to the wider practice of design, one feature of which seems to be a particular difficulty for many engineers, namely that of aesthetics, which seems subjective and not open to rational analysis. It figures quite predominantly in the wider field of visually-led design, ranging from architecture through to graphic design. This general field of design studies has been quite a fruitful area of philosophical, or at least methodological, investigation since the early 1960s and certain engineering designers have contributed significantly in the field of design research, which could be said to have been started in 1966 by the chemical engineer Sydney Gregory [2].

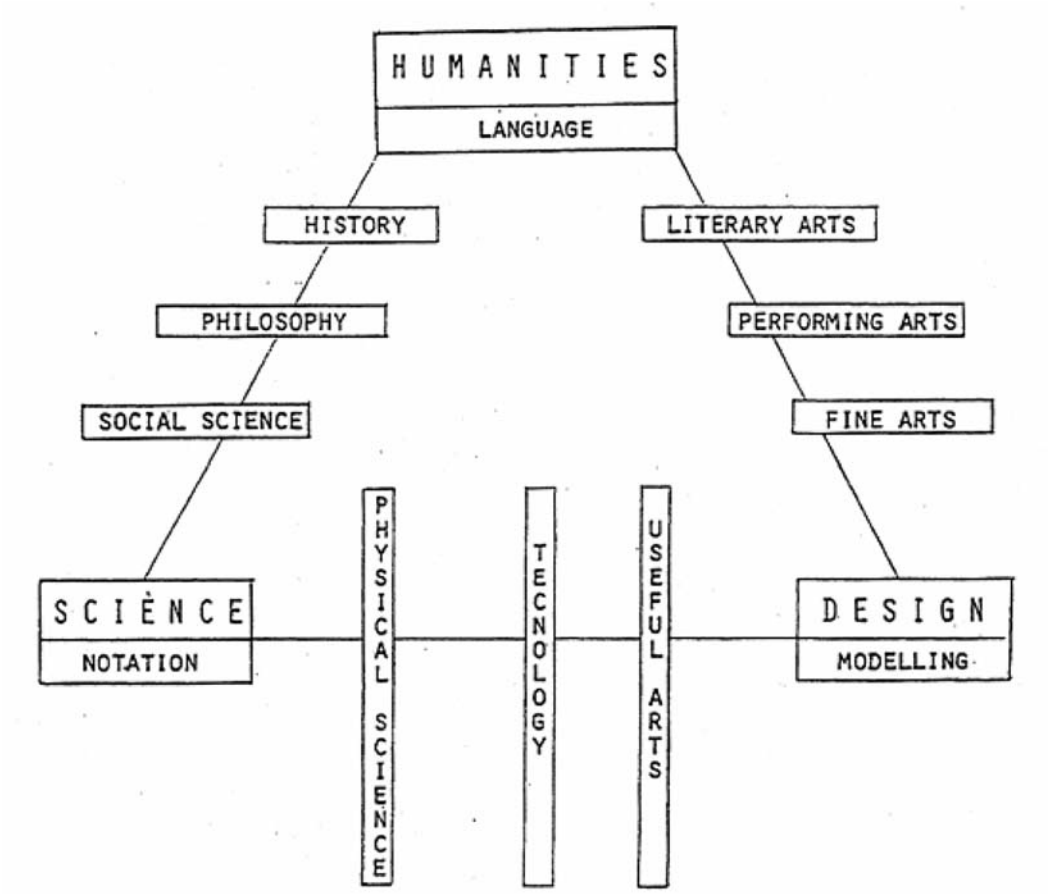


Figure 1. Bruce Archer's representation of design as the third culture [3]

A particular view was put forward in 1979 by the then Professor of Design at the Royal College of Art, Bruce Archer, who saw design as a third culture, alongside the humanities and science (see Figure 1). There was design - not *engineering* design, but design - with modelling as its means of communication, as distinct from language, for the humanities, and notation, for the sciences, which goes beyond mathematics to cover other notations as well. Archer's diagram is provocative, and I would go further and argue that *engineering* design should be placed at the centre of his triangle, and possibly uniquely so, since it requires not just visual models and mathematical science, but also verbal language. If engineering design is being adequately practised, the designer must be adept at all three of these means of human communication.

Archer's emphasis on communication through models leads me on to consideration of the use of visual representation to describe the design process. It is interesting that philosophers of science, as argued by Peter Lipton in the first paper in this volume, even when talking of the scientific method do so using verbal language, or occasionally logical notation, whereas design methodologists usually produce diagrams to explain the design process. This is done not so much to explain how design is performed but rather to describe the process one appears to undertake in designing something. I am a ship designer and we, as a discipline, are very attracted to seeing the design process as iterative (ie. the design spiral). This the inherent interrelationship of design parameters necessary to reach a balanced design solution [4].

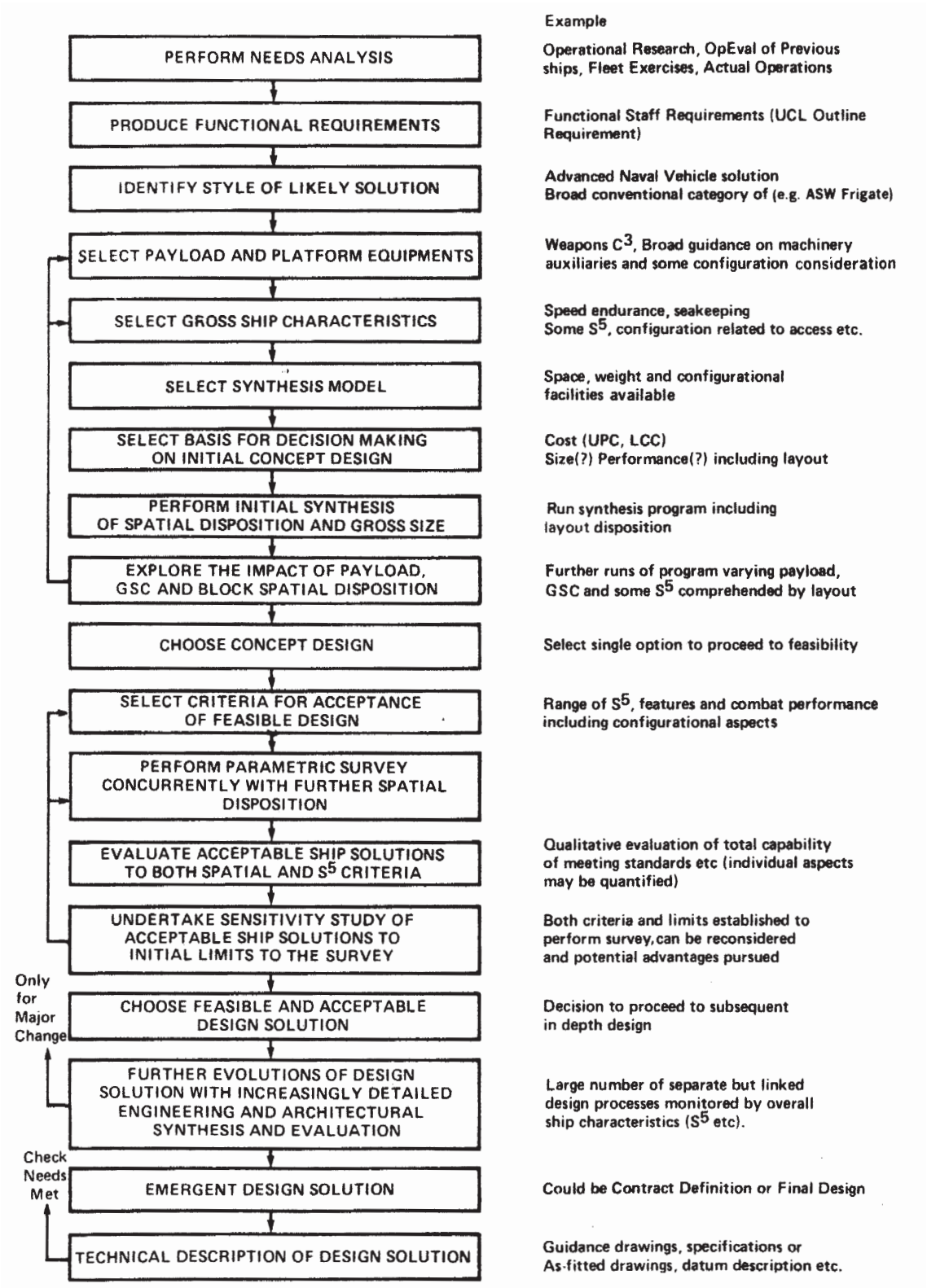


Figure 2. A representation of decision making in the ship design process [4]

In contrast, figure 2 is a sequential representation of the ship design process, listed as a series of significant decisions, rather than just a schedule of activities where each step would be distinguished by its demand in time and effort. In fact, most of the resources (time and money) expended in the process of designing a ship actually take place in the last three steps of figure 2. More importantly, very early on, many crucial decisions are made and that is really what this diagram is about. In that respect, such models can be useful - at least in describing the critical early steps in design. To reinforce Ferguson's insight about the myriad small decisions

in design, step eight in figure 2 can be considered further given it is a rather a significant step, being the production of an initial sizing or design synthesis. As figure 3 shows, that step itself can be broken down into a series of steps even though it is just one step inside the overall process representation. It also enables me to make a crucial point, one which is very often not mentioned when such models and descriptions of the design process are presented.

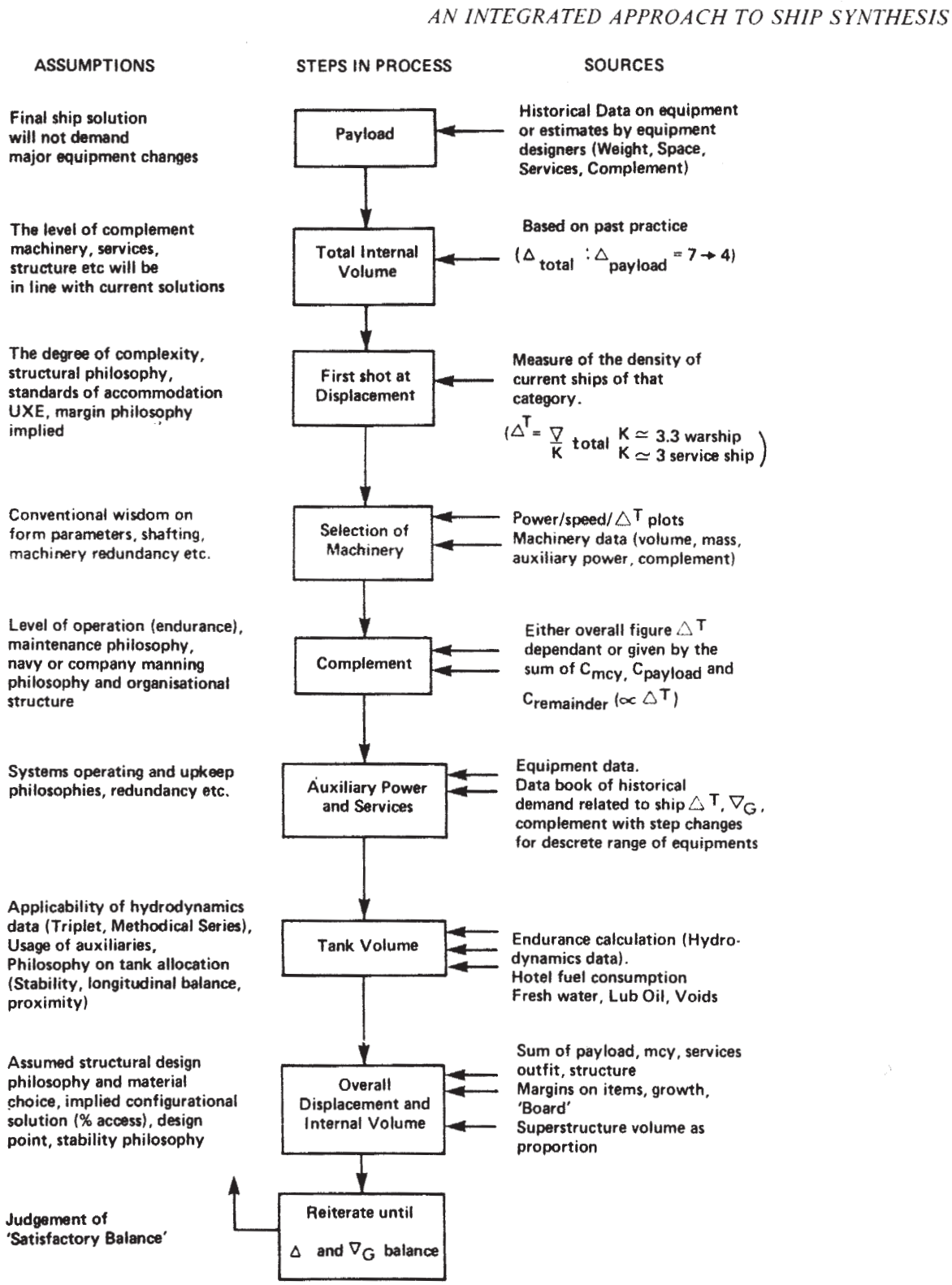


Figure 3. A initial ship sizing sequence showing assumptions and data sources [4]

This description of the sequence to numerically size a new ship design actually has a host of implied assumptions. Without reading all these specific examples, one can see, at every single step, there are several assumptions, which can be significant. Worse than that, each step requires specific sources of information to achieve that step in the process. What I want to draw attention to is that, while such a process is ordered and sequential, it involves a lot of assumptions and choices of information, which the engineering designer is, hopefully, aware that he or she is consciously selecting. It is also often the case that much of this information is in the form of simplified rules of thumb or previous practice.

Now there are many models of the engineering design process and it seems to be the case that everyone has his or her own. My message is that in using such models of the process designers need to use them with care and be very aware of their inherent assumptions and constraints. Furthermore, drawing any general insights is inevitably fraught with qualifications and exceptions.

The range of engineering design practice

One particular qualification in any methodological analysis of engineering design, that I consider important to make, is that there is an immense range of design practice under the umbrella of engineering design. Often textbooks on engineering design, in describing engineering design in general, are actually referring to the design of engineering components. In a way this is reasonable, given that larger and more complex engineering products themselves consist of such components. However, I would argue that the design process model for the mass produced component or the small engineering product manufactured in very large quantities, is inappropriate for the other extreme of engineering practice, which I shall come to presently.

Figure 4 is a general representation of engineering design produced for the German engineering industry as a systematic procedure by V. Hubka [5]. It is a useful model but it is really specific to product design and could be considered applicable to larger manufactured goods and vehicles, such as cars or even, possibly, aircraft. All these products are typified by the fact that they are evolved using physical prototypes - often many prototypes and usually at full scale. This ability to resort to prototypes, rather than just drawings, models, mock-ups or even computer models, is a great means of risk reduction. It also usually assists with tooling up a factory for mass producing the product. The designer still has to create a synthesis but, through prototype production, there can be seen to be a particular design process not dissimilar to Hubka's component design process.

Admittedly, at the mass manufactured extreme, a modern aircraft is obviously highly technically complex, but what I would like to denote as physically large and complex systems differ from the mass factory produced product in that these complex products tend to be produced in ones or in small numbers. Thus they lack both prototypes and the resultant design of the manufacturing production line, to achieve the classic 'fly before buy'.

A modern version of non-prototype, one-off complexity could be the design of software for major engineering projects, including aircraft, powerstations and large scale communications, command and control systems. Although very aware, from managing naval ship projects, of the significance of software development in project risk management ("it always goes wrong"!), I have tended to regard software design as the prime justification for adopting a systems engineering approach in engineering design. So I was intrigued by Richard Coyne's assertion, from the stance of an IT practitioner, that there are no formulaic theories of design, just generalisations [6]. This would seem to undermine not just systems engineering but also the endeavours of the design methodologists, as does Coyne's championing of metaphor over models as the basis of design.

I now come to what I would call complex and *physically* large scale design, which is sometimes called 'bespoke' or 'made to order' design. This is typified by civil engineering constructions, large process plant production, ships and offshore facilities. Given such products are large and often one-off, they do not have prototypes, nor are their subsequent manufacturing facilities re-designed for each new product run. Rather they are assembled on site or in a piecemeal manner. This has significant consequences for the design, especially if the design product also has to accommodate human habitation for considerable periods (not just short term transportation of people, as in cars and planes). This then is almost whole system design.

Because these products are few and immensely expensive, they tend to be produced for long usage and hence adaptability often figures as a design objective, as can multifunctional usage. Thus, at the front-end of the design process, without the comfort blanket of a prototype, working out what is wanted and what is affordable can be a complex process. Interestingly, the design theorists have coined a term, called 'the wicked problem', to encapsulate this issue, namely that determining the requirements can actually be more challenging than the subsequent design work to meet those emergent requirements [7].

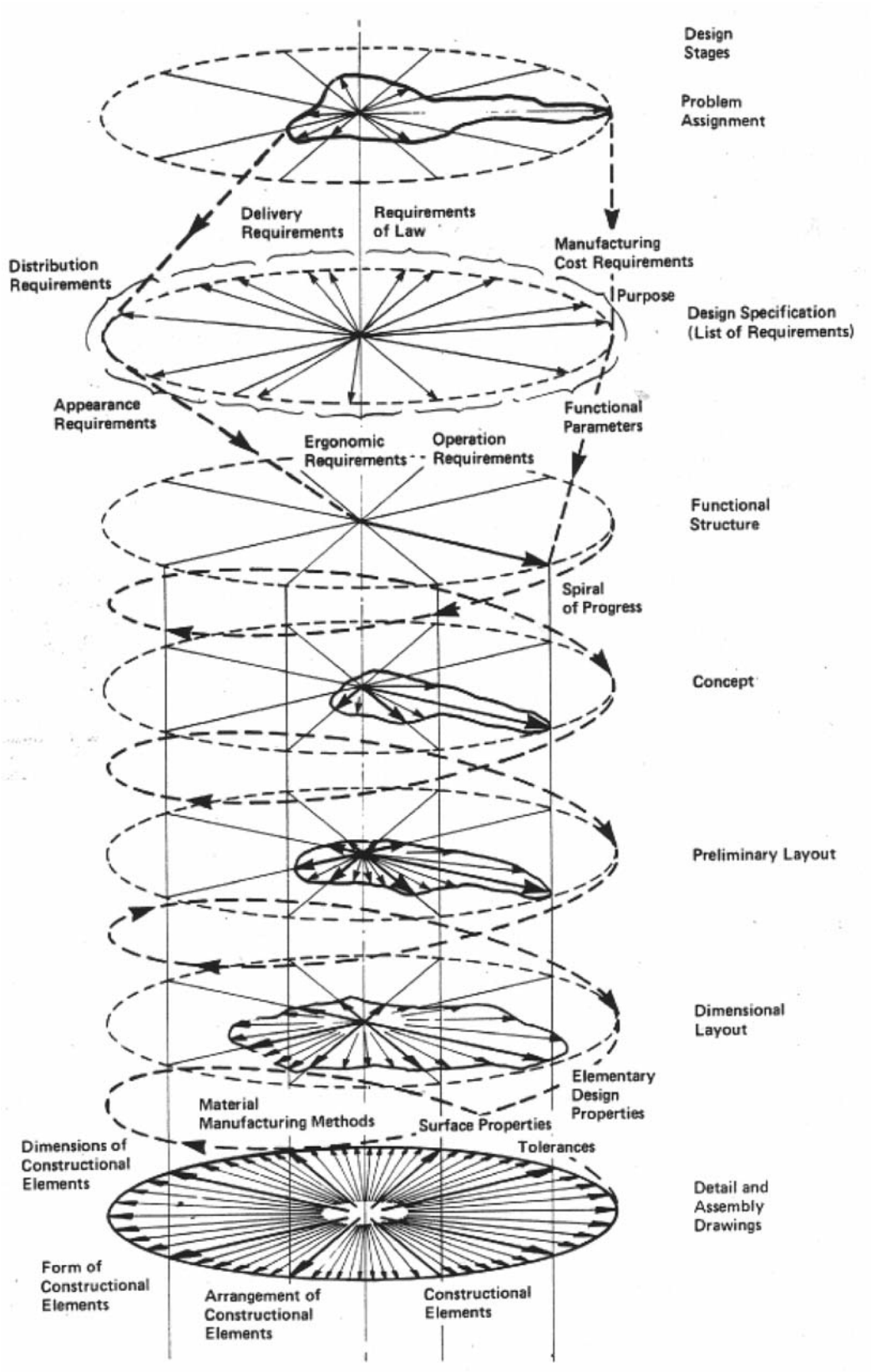


Figure 4. A model of engineering design due to Hubka [5]

This distinguishes the design of such complex products from the many descriptions of the design method that start off with a clear need or set of requirements, (see figure 4). Thus, for such complex designs, the initial phase is one of elucidating the requirements. That is a trial-and-error dialogue between the designer, producing prospective solutions (and the plural is very important here), and the customer/owner/end-user, such that both are involved in trying to reveal what is really wanted and achievable. However, some customers fail to appreciate the importance of this process [8].

Having said that the design of complex large scale products has these characteristics, I want to further complicate the picture by pointing out that actually there is in this category of design a range of design practice, as shown by table 1. I have tried to use a set of examples not wholly taken from my own field of ship design. The table shows that the complexity of the process can range from simple modifications of an existing product, through to radical configurations, which largely adopt the current technology but are still rarely attempted due to the risk of unknowns (without the prototype), and then even more rarely radical technological solutions. These latter have to be designed in a manner much more akin to aerospace practice, both in requiring vast development costs (including prototypes) and the need to design and build specific tooling and manufacturing facilities. So any attempt to systematise engineering design practice must at least recognise that there is this spectrum, from component design (figure 4) right through to a range of ways of approaching complex design (table 1).

Type	Example
second batch	A stretched Airbus
simple type ship	Most commercial vessels
evolutionary design	The three generations of current RN nuclear submarines
simple synthesis	Type 23 RN Frigate concepts
architectural synthesis	UCL Mothership studies
radical configuration	The London Millenium Bridge
radical technology	The Space Station

Table 1. Complex design categorised by increasing design novelty [9]

Some philosophical issues in engineering design

This essential preamble has thrown up some clear methodological, if not philosophical, concerns. I would therefore like to mention, very briefly, some of what can be seen as issues in the practice of engineering design that have a philosophical dimension to them and which we, as engineers, disregard at our peril.

The first of these is how, at least for complex, large products, one can synthesize a new design. In the first of these papers Peter Lipton spoke as a philosopher of science and pointed out that such philosophers really have little to say that help us understand how you start a new design. Thus Popper seems to see the nearest scientific equivalent, the creative process of scientific conjectures, as more an issue of psychology than philosophy [10]. Also, the endeavours of design researchers to study design synthesis, seem largely to be so artificial, small scale and usually irrelevant to the practice of complex design, or in trying to represent the real world in a student design exercise environment, they give very little insight.

It is possible, however, to draw on two studies that were done in the early days of design research. One was by Jane Darke [11], who quizzed architects, rather than engineers, on how they came up with new buildings. The indication was that an architect searches for a key design feature or generator to provide the basis of the new concept. Interestingly, I have found some parallels from comments by designers of post-Second World War British naval ships, where they seemed to adopt a similar approach: they looked for some key aspect that would generate the new design for them [12].

The other study was by Janet Daley [13] whose insight into the creative process suggested that individual designers bring a set of schema to their design creation. The ones she identified were (unsurprisingly) visual; verbal, which brings us back to my comments on Archer's aspect of communication in engineering design; and lastly, very interestingly, value systems, which echoes John Turnbull's comments in the preceding paper. If these two pieces of research are plausible, then they have some quite profound implications for the all-pervasive application of computers to engineering design.

I believe there are philosophical issues with regard to both the management of the engineering design process through systems engineering and the virtual obsession many engineers have with applying numerical optimisation to design. Specifically engineering designers need to cultivate a healthy scepticism respecting rational optimisation techniques, often applied without full justification, and also to a wider over-reliance on quantification of, often, disparate issues. This healthy scepticism is essential if engineers are to avoid dubious compounding of "apples and pears" or even the over-application of genetic algorithms.

Something that many advocates of systems engineering and simple models of the design process seem to be falsely wedded to is the functionalist philosophy, despite the fact it has long been recognised by architects as just another style. Let us be quite clear: Form does *not* follow Function. Mies van der Rohe even reversed Sullivan's epigram, hence his pioneering of open plan (adaptable) high-rise offices. You can't have a dialogue to solve the "wicked problem' of requirement elucidation through a process that describes requirements in "functional" (i.e. non-material) solution abstractions. Furthermore, the major concern of the customer is bound to be affordability (unless you are lucky enough to be designing a billionaire's mega-yacht, where form is everything) and that affordability is only accessible through prospective material solutions, not (seemingly abstract) functional statements. So recognition that an important decision is the style of the prospective solution is part of any design methodology (see the third step in figure 2).

Two other related issues, arising from the reliance on the computer for complex product design are those associated with the need to tackle the integration task and with the ability to judge the output of complex computer software. Integration requires a sense of the total system, as well as the individual details. Of course, there is the engineering truism that the 'devil is in the detail' and most engineers want to get into the detail because they know that is where failure is likely to occur, even if the source may be in the overall decision process. So you also have to be able to see the totality as part of the integration and if it was easy we would not all be so fascinated by it. Computers can now provide the most wonderful output as part of (say) a finite element analysis but how representative is the input? How appropriate is it to the actual problem? There are big philosophical and educational issues to be tackled.

So what do all these issues mean for engineering design practice, given the ubiquitous presence of electronic computation, not just for number crunching analysis but, thanks to computer graphics, for CAD, Simulation Based Design and Virtual Reality? I have a very strong belief, born not just of practice in ship design but also involvement in the research and development of computer aided preliminary ship design tools, that the crucial word in CAD is *aided*. Thus any recourse to "black box" CAD systems or optimisation routines that leave the human designer as little more than a key board operator, unaware of the basis of the synthesis process and the decision making of his or her, so called, design tool, has to be strenuously resisted.

Do we need a philosophy of engineering design?

There are two facets to this question - design and engineering. Design methodologists (typified by discussions in the journal "*Design Studies*") have over the last twenty years explored the philosophy of design. In a special issue in 2002 on the philosophy of design, Galle as editor concluded that the philosophy of design was an immature union of philosophy and design research in "the pursuit of insights about design by philosophical means" [14]. So if philosophical practice is concerned with "the principles underlying any sphere of knowledge" [15], this would all seem to come down to investigating the principles that lie behind our practice of design. The issues of function and form creation were prominent in the articles, in that special issue, as were the issues of demarcation and taxonomy.

The second element is of course engineering and this seminar and the others in the series are, I believe, a clear recognition that the discussions on a philosophy of engineering - the principles behind our practice - is over due. I hope also the previous section's brief outline of some of the issues relevant to the future practice of engineering design are convincing evidence that there are some important principles to address as part of the Academy's endeavour. There would seem to be good reasons for supporting the Academy's desire to address at a philosophical level the nature of our business of engineering and, specifically, engineering design. Engineering still lacks the status it ought to have and fails to attract and retain the brightest young people in sufficient numbers and one element in that is a certain feeling/belief of intellectual inferiority *vis a vis* the sciences. Understanding what we are doing and articulating this with an intellectual rigour that conveys the worth and excitement of our discipline, could counter this misapprehension. Such an endeavour needs to be done without simplifying and straitjacketing the practice, as I hope the distinctions in engineering design, I have been careful to articulate, have highlighted. So striving for better understanding of the principles behind our discipline is a totally justifiable and important objective.

So what do we want from a philosophy of engineering design? Without being prescriptive and trying to reflect the creativity and sophistication of that practice, I have pushed at the front end of complex engineering design for a more inclusive approach to computer aided design. This has introduced at the initial design synthesis a representation of the physical architecture of the artifact, as a set of three dimensional building blocks [9]. In so doing the impact of the design building blocks on the design is integrated with the existing numerically based synthesis, summarised for ships by figure 3. This is a way of fostering a designer-led approach that is creative and holistic in engineering design terms. The approach recognises the complexity of designing engineering products, as can be discerned from figure 5 and references 4 and 9. Like most engineering practice, to produce such designs requires domain knowledge and experience; it is thus a challenge but also provides a more philosophically satisfying approach to complex design.

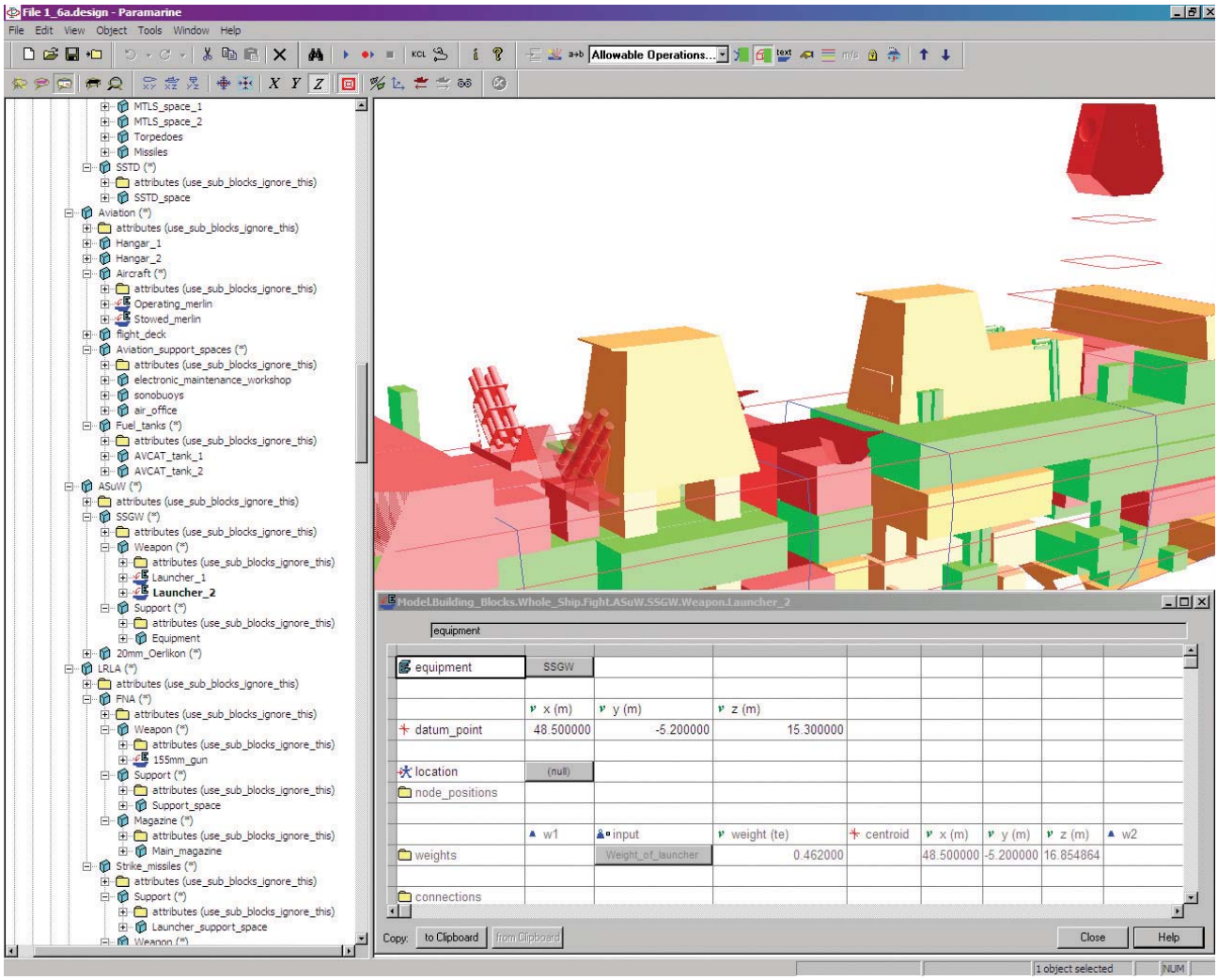


Figure 5. An example of the design building block ship description using the SURFCON design tool and showing the graphical screen, the object oriented hierarchy and an example of initial ship design analysis to achieve a balanced ship design [9]

References

1. Ferguson, E. S. (1992) *Engineering and the Mind's Eye,* Camb. Mass: MIT Press
2. Gregory, S. A., (Ed.) (1966) *The Design Method*, London: Butterworths
3. Archer, B. (1979) 'Design as a Discipline- Whatever Became of Design Methodology?', *Design Studies*, Vol. 1, No 1
4. Andrews, D. J. (1998) 'A Comprehensive Methodology for the Design of Ships (and Other Complex Systems)', *Proceedings of the Royal Society, Series A* (1998) 454, p.187-211
5. Hubka, V. (1982) *Principles of Engineering Design*, Translated Eder, W. E., London: Butterworth Scientific
6. Coyne, R. (1995) *Designing Information Technology in the Postmodern Age - From Method to Metaphor*, Camb. Mass: MIT Press
7. Rittel, H. M. J. and Webber, M. W. (1973) 'Dilemmas in the General Theory of Planning', *Policy Sciences*, Vol. 4
8. Winter, D. C. (2007) 'Getting Shipbuilding Right', US Naval Institute Proceedings
9. Andrews, D. J. (2006) 'Simulation and the design building block approach in the design of ships and other complex systems', *Proceedings of the Royal Society, Series A* 462
10. Lipton, P., 'Engineering and Truth', (chapter 1 of this volume)
11. Darke, J (1979) 'The Primary Generator and the Design Process', *Design Studies*, Vol 1, No 1
12. Brown, D. K. (1983) *A Century of Naval Construction*, London: Conway
13. Daley, J. (1982) 'Design Creativity and Understanding Design Objectives', *Design Studies*, Vol. 3, No 3
14. Galle, P. (2002) 'Philosophy of design: an editorial introduction', *Design Studies*, Vol. 23 No 3
15. Chambers Twentieth Century Dictionary, London 1971

6. Roles and Rules and the Modelling of Socio-Technical Systems

Professor Maarten Franssen
Delft University of Technology

Maarten Franssen is associate professor at the section of Philosophy in Delft University of Technology. He studied theoretical physics and history and received a PhD in philosophy from the University of Amsterdam on a study of the foundations of the social sciences. His research interests include the application and integration of social-scientific and philosophical approaches and theories in engineering design and the engineering sciences.

I am a philosopher at the Section of Philosophy of Delft University of Technology, where I participate in a research project on the design and modelling of socio-technical systems. This project involves five people, most of them philosophers by profession but with a training in science or engineering. As a philosopher, my contribution to this workshop looks at engineering from a philosophical point of view, driven by philosophical interests, but philosophers would consider this research project technical rather than reflective, meaning that it is focused on a clearly identifiable practice taken more or less in isolation and that it aims at a clear and precise definition of concepts and a formal treatment of the relations among these concepts.

Although I moved into philosophy from an education in theoretical physics, I am also a historian by training, and I will start with a historical introduction to my topic. Around the turn of the seventeenth century, the leaders of the Dutch army, which was fighting a war of independence against Spain, introduced a major revolution in the way an army was supposed to go about its business. Certainly in those days, the armies consisted primarily of professional soldiers whose fighting had little to do with defending their families or their country. A major difficulty the generals of these armies had to solve was to get their soldiers to fight the enemy instead of running away. They used various methods to arouse their soldiers into a frenzied state of blood thirst, but this was not very reliable. The revolution introduced in the Dutch army, which consisted almost entirely of German soldiers, was to rely instead on discipline and training. Experts analysed the handling of a particular weapon by a soldier into a series of distinct basic actions, and a soldier was trained to perform each of these actions in a machine-like way in response to a corresponding command. The following is the full list of commands for the 43 movements into which the firing of a firelock by a musketeer was analysed, such that at the end of the sequence the soldier has fired his shot and is ready to go through the cycle again and fire a second shot:

> Put your firelock to your shoulder and march. Take your firelock from your shoulder. Keep it upright with your left hand. Hold your firelock in your left hand. Hold your fuse in your right hand. Blow your fuse and keep it well. Apply your fuse. Try your fuse. Blow your fuse and open your pan. Shoulder your firelock and take aim. Fire! Lower your firelock and keep it in your left hand. Remove your fuse. Hold it between your fingers. Blow your pan clean. Put powder on your pan. Close your pan. Shake-off your pan. Blow your pan clean. Turn your firelock. Lower it along your left side. Open your ball case. Load your firelock. Pull out your ramrod. Hold your ramrod at the end. Stamp your powder. Pull out your ramrod. Hold it at the end. Put your ramrod back. Bring your firelock forward with your left hand. Hold it upright with your left hand. Put your firelock over your shoulder. Keep your firelock balanced on your shoulder. Take your firelock from your shoulder. Lower it with your left hand. Hold your firelock fast. Hold your firelock just with your left hand. Take your fuse in your left hand. Blow your fuse. Apply your fuse. Test your fuse. Close your pan. Stand prepared.

The innovation dates back to the late sixteenth century and was introduced by Maurice Prince of Orange, who was the stadtholder and supreme military chief of the Dutch army in the war against Spain. Among his staff were a great number of engineers, among them Simon Stevin, the 'Dutch Galileo' (who we know did some of the things that are perhaps falsely ascribed to Galileo, such as dropping two objects of different weight from a tower to see whether they touched the ground at different times, as Aristotle predicted). And indeed this innovation of replacing the traditional frenzied band of soldiers by an ordered group of thoroughly trained professionals is in my opinion a product par excellence of the engineering approach to the design of complex artefacts.

A crucial aspect of this approach is that it takes the behaviour of the overall system to follow in a determinate way from the behaviour of the system's components, and this is regardless of whether these components are lifeless material objects or human beings. The soldier is human, his firelock is mechanical, but both are treated in essentially the same way. The components of the firelock behave in certain ways in reaction to the mechanical forces applied to them, resulting in a (reasonably) predictable behaviour of the whole firelock. Similarly, provided he is suitably trained, the soldier reacts completely deterministically in response to constantly changing input. To each type of input - a particular command - he answers with the 'right' output - a particular basic action - and this output ensures that the whole system - the army - functions as designed. The movements of the soldier reflect a kind of algorithm that makes the soldier go through the whole series, making the actions of the soldier fit in perfectly with the system for which this 'element' was designed.

I have just now casually introduced the notion of a system, in particular an engineering system, being a system designed, realized and operated by engineers to achieve a particular purpose. This term 'system' is in itself not very informative. It was introduced into engineering in the 1940s, leading to the rise of systems engineering in the 1950s and 1960s, but at the same time obtained a central place in biology as well. The common understanding in this literature is that a system is a complex whole consisting of elements or components that are related to each other. This makes almost any technical artefact a system. The research project I am participating in is interested in a particular kind of system, a kind for which the name socio-technical system is now increasingly becoming standard. Socio-technical systems are hybrid systems in the sense that their components come from (at least) two quite different categories of things: some components are ordinary material or hardware objects, whereas others are human beings. Some researchers would say that most socio-technical systems also contain elements from a third category, a category consisting of abstract entities like rules and norms, but this is a point I will not discuss today. Anyway, on both views the Dutch army with its disciplined soldiers was a socio-technical system.

In general a clear distinction must be made between two ways that humans can be 'involved in' engineering systems. One way is that of being an operator within the system, the other of being a user of the system. From the point of view of the army as a whole, every individual soldier is an operator of a tiny subsystem - a single firelock - of the overall army system. The officer issuing the commands is an operator of a subsystem at a higher level, a company of musketeers, which consists of hardware and operators. Operators are always elements or components of the system. The system requires their operation activity to perform its function and realise the purpose(s) of its user(s). Users of the systems, in contrast, ipso facto do not belong to the system that they use.

Since a model of a system should contain all elements that are relevant to the functioning of the system, a model of a socio-technical system must include the operators. In the literature, two approaches can be distinguished as to how this is to be done. The first of these is often called hard-systems thinking, the second soft-systems thinking. Hard-systems-thinking was the predominant, in fact the only form of systems analysis until the early 1970s. In that period the emphasis shifted from hard-systems engineering to soft-systems engineering.

Hard systems are analysed through their various components, the way they behave and the way they hang together. The general idea, and the introduction of the notion of feedback, was a major innovation in the early 1940s and 1950s. Figure 1 shows two examples. At the left is a simple thermostat, consisting exclusively of hardware elements, and to the right there is another example which is seen to be modelled using the same general idea, but now including a human element - the 'manager' - instead of the hardware control unit of the thermostat. The human element, nonetheless, is supposed to function exactly as the technical elements and it is accordingly conceived as an algorithmic or completely deterministic process, like the Dutch musketeer in my introduction.

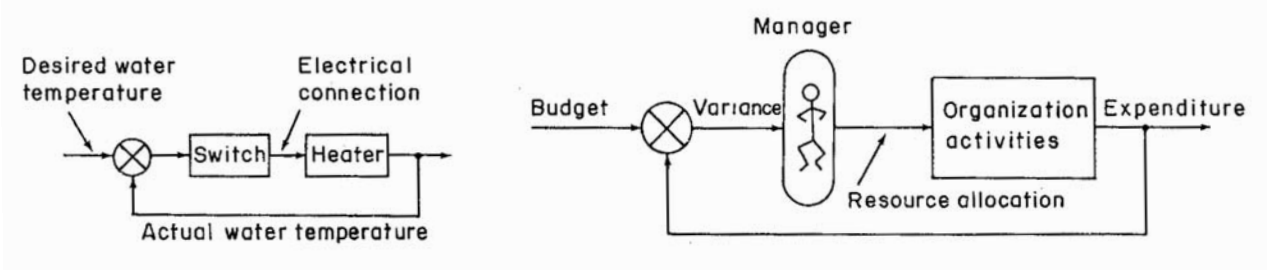


Figure 1. Two examples of hard-systemsmodelling [1]p.6

In soft-systems thinking, the conception of what the system consists of has radically changed. The hardware elements that are predominant in the hard systems approach have completely dropped out. The system consists purely of people being interrelated through social ties. Figure 2 shows a model of what Brian Wilson calls a human activity system, which he distinguishes from a design system. A design system is exclusively focused on the hardware build-up, whereas the human activity system is exclusively focused on the human build-up. What is being modelled are roles that are related in a certain way to each other, and this system of roles is grounded in a social system which consists of human beings connected through social relations. That these human beings are also 'hard' beings of flesh and blood is immaterial. Likewise, the activities modelled in the left part of figure 2 will usually be activities that involve the manipulation of hardware, but what exactly is manipulated and how whatever is being manipulated performs its function is not included in this conception of a system.

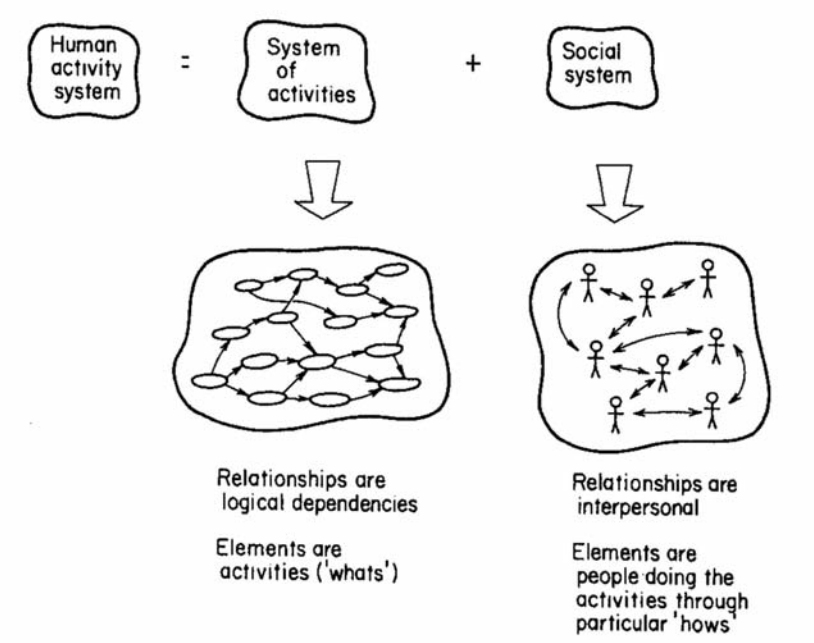


Figure 2. An example of soft-systems modelling. [1], p. 28

Originally systems engineering was all about hardware systems. The basic task of the systems engineer was to ensure that the physical behaviour of the various components was perfectly coordinated, to ensure a proper functioning of the whole device. When these systems occasionally contained humans as elements, this was because it was too difficult or too expensive to build hardware elements for certain subfunctions, and these were therefore left for humans to perform. From this systems-engineering point of view, which saw any system as a coordinated arrangements of components characterisable by input/output schemes, any place taken by a human operator or user will be characterised in terms of a determinate input/output scheme. Hard-systems thinking is based on the assumption that any human subsystem will actually perform its function - by instruction or training or perhaps still other means - in accord with the system design. However, the modelling techniques, which presumably had to come from the social sciences, to predict whether or not the human subsystems would perform their functions according to the system design, and to what extent in what circumstances, were largely absent. But instead of developing the required knowledge, systems thinking turned its back to the sort of systems that engineers are designing and, in a sense, changed the subject toward the modelling of only the managerial and organisational aspects of systems.

This, obviously, has not made the original problem disappear; far from it. Large-scale, complex systems - typically infrastructures like air and road transport systems, public utilities and telecommunication systems - having numerous close connections between what humans do and what happens at the hardware level, are of enormous importance in modern society. Within these systems there are operators at many levels who continuously have to make decisions that are connected directly to buttons that are pushed or switches that are pulled, and their decisions have immediate repercussions on the hardware level, where they affect the lives of millions of people. The engineers who are involved in the design, implementation, operation and maintenance of these systems must, therefore, have substantial knowledge of these human subsystems, since the actual performance of the system depends on the ability to control the performance of these operators and to foresee exactly, or at least sufficiently exactly, what particular sequences of decisions can be expected, how likely or unlikely they are depending on the circumstances, and what will be their consequences.

Can the social sciences deliver here what the natural sciences have delivered in the past concerning our control of hardware systems? An important contributor to current ideas on modelling systems according to the views of soft-systems thinking, Brian Wilson says (in [1] p. 23):

> Since our concern is with the development and application of a systems language to complex problem situations involving people, some contribution from the activities of social science and sociology could be expected. However, these disciplines have not yet produced much in the way of methodologies that can actually be used in the analytic sense, though concepts related to perceptions and meaning, values, roles and norms have proved to be useful and have made a contribution to the particular language involved in the use of human activity systems.

Wilson's observation is, I think, largely correct. He certainly articulates a view that is widely shared among engineers. Nevertheless it seems to me that applications of certain types of knowledge from the social sciences to the design of engineering systems can be found. A brief look at some examples will also show what sort of knowledge is missing, and what sort of knowledge we additionally need for the design of socio-technical systems.

First, however, it should be noted that the social sciences do not by far employ a unified conception of human beings. There is rather a whole spectrum of conceptions. At one extreme there is a more or less biological conception that comes very close to the law-like processes with fixed input/output relations favoured by hard-systems engineering. At the other extreme, there is a view of humans that more or less answers to the way we look at other people in daily life, where we treat other people as similar to us in having ideas and in being motivated to do certain things on the basis of these ideas. According to this view, people have beliefs and expectations about the world, they have goals and aims, they find some of these goals more important than others, they rearrange their beliefs and their goals while learning from their experiences, and they have a certain knowledge of what will happen if you do this or that. Most importantly, with a view to the modelling of socio-technical systems, is that according to this view, when people face a decision - which of various possible courses of action to take - they have an interpretation of the situation in which they are facing this choice, and in making their decision they take into account what they consider to be the consequences of each course of action, given their interpretation of the situation.

These are two quite different models of people, but they can both be found everywhere in the social sciences. The latter model, of a human being as an intentional being, to use the philosopher's term, is - idealised to a great extent - the dominant model in (theoretical) economics, whereas the former model, that sees humans as systems with specific input/output characteristics, is in (social) psychology and large parts of sociology.

This input/output model certainly finds application in engineering systems. One example is the design of rooms which may have to be evacuated in emergency situations. If there are only a few doors through which the occupants of the room must leave, which is the most common situation, what you see is that you get a 'clotting', so to speak, of people who all want to get out through the door at the same time and in this way make it much harder for anyone to leave the room. This seriously delays the evacuation of a room during emergencies. Now it was derived from fairly simple input/output models of how animals move in the immediate presence of other animals that if you place an obstacle right in front of the door, you achieve a much smoother and faster emptying of the room. This is not a solution you would easily have thought of if your frame of reference is that of a conscious person picturing the way from his or her own position to the exit door and planning the fastest route to get there.

Another example is the (more or less) famous mock fly in the urinals of gents' toilets, first used at Schiphol Amsterdam airport, I believe. (I hope this example is not a breach of protocol. Philosophers, however, are not ones to shy away from the brute facts of human existence.) The idea was that in order to increase the cleanliness of the toilets, you need to assist the men who use them in taking better aim. Simple empirical evidence suggested that something like this fly would work. I was told that this was discovered in the military, where you have soldiers camping out for days in rough terrain and one or two soldiers are responsible for digging out the facilities and keeping them in good order. It was discovered that it helped enormously if you put an empty tin can in the middle. Again, what is applied here is a simple empirically discovered input/output model of humans, or at least the male half of them. There is no need to establish that men actually aim for the fly, or take a decision to do so. They may never give the matter any thought, they may even deny vehemently that their behaviour is in any way affected by the presence of the fly. What matters is that there is this positive effect.

These two examples show some elementary behavioural characteristics of people that are made use of in the design of some extremely simple artefacts. These elementary forms of behaviour are, however, not what we are after for the design of complex engineering systems. Such systems consist of many hardware subsystems, each often quite complicated in itself, and the overall system functions properly only if the behaviour of the various subsystems is precisely co-ordinated. Certainly in the early days of systems engineering this proved to be possible only by introducing humans as operating subsystems. In modelling the whole system, engineers conceive of these operators as similar to the hardware elements they are accustomed to and know how to model and control. Indeed, engineering opinion will have it that overall predictability is possible only through predictable subsystems, irrespective of whether these subsystems are hardware devices or people.

Figure 3 gives an example from the very authoritative ISO standard for systems engineering. It shows a model of an aircraft system as part of a larger air transport system, which is again related to other transport systems.
The characterisation of the various subsystems that make up the aircraft system does not indicate in any way whether or not some of the operating and controlling is done by people instead of machines. At the extreme right of the aircraft system model, the crew, which is an all-human subsystem, is presented as completely on a par with the other all-hardware subsystems. The air-traffic control system in the enveloping air transport system is operated by people - air-traffic controllers -supported by hardware systems, but it cannot be seen from the model that this is so, or that this is so for the air-traffic control system but not for, e.g., the ticketing system, which could be a completely automated subsystem.

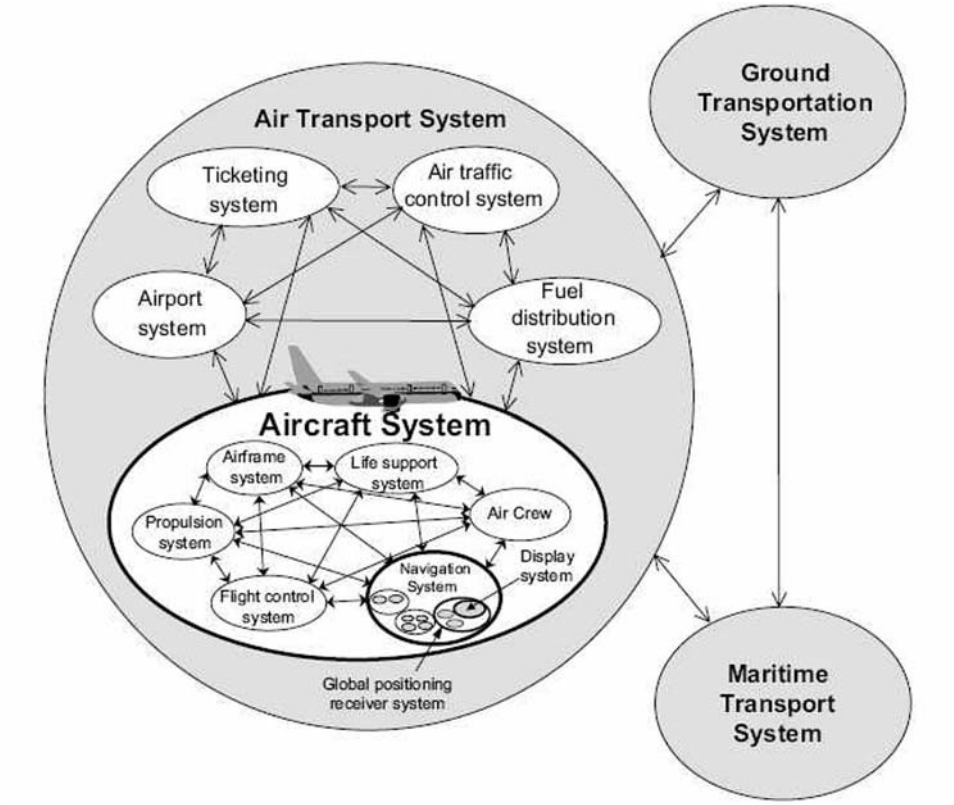


Figure 3. Model of an aircraft system as embedded in an air transport system, which is related to other transport systems. (From[2] p. 53)

Of course people cannot be designed on the basis of a thorough knowledge of all relevant laws of nature and then manufactured accordingly, as hardware devices are. This is how engineers are able to guarantee the predictability of hardware devices in specified conditions of the environment. For people engineers try to come as close as possible to 'designing' a subsystem with a specified input/output characteristic by subjecting people to rigorous training (like the Dutch musketeer) or by furnishing them with an exhaustive list of circumstances that may arise and the required action to perform in case the circumstances arise. In this way it is justified to model them as deterministic subsystems like any hardware device.

Especially in the design of socio-technical systems, however, failing to take into account the fact that human beings are fundamentally different from machines will create serious difficulties. The point is not that humans as biological organisms are fundamentally different from machines. That remains to be seen. The point is also not so much that people make mistakes, by which I mean that they choose the wrong action when a particular condition materialises, or that they fail to recognise a condition as one where they should take a particular action, and this is admitted by the operator. Hardware malfunctions can also never be ruled out, due to our incomplete knowledge of nature. The point is that people can contest a judgement that actual circumstances are or were precisely equal to a specified condition in their list of instructions, and can therefore contest whether they ought to choose or should have chosen a particular course of action. People do not coincide with the roles they fulfil. They are defined, rather, by their goals and desires,

their beliefs and expectations, as individual persons. Their judgement will, therefore, involve a broader range of considerations than any list of instructions will contain. Finally, people are as individual persons part of a social system. They have responsibilities, both in the roles they fulfil and as individuals, and they are held responsible for their deeds. This seriously affects which courses of actions they will choose.

I will illustrate the relevance of this issue by presenting as a case study an analysis of an air-traffic accident that happened on 1 July 2002, when a Russian passenger aeroplane collided in mid-air with a DHL mail carrier near the town of Überlingen in Southern Germany, resulting in the death of all people aboard the two aeroplanes. Figure 4 is a schematic representation of the situation in which the accident occurred. The two upper squares represent the two aeroplanes involved. The left square is the Boeing 757 operated by DHL, the right square is the Tupolev-154 from Bashkirian Airlines. Both planes are flown by a crew, who control the plane through all kinds of hardware devices (which I have indicated by PCD or plane-controlling devices, not a standard term). Both aeroplanes are subsystems in a larger air-transport system, which contains a higher-level operator system, air-traffic control (ATC), represented by the lower rectangle in figure 4. Air-traffic control also consists of a human operator connected to an array of hardware systems.

Apart from the two crews and the air-traffic controller, there was a third, independently defined operating subsystem present in the situation which consisted entirely of hardware. In response to earlier mid-air collisions, all commercial aircrafts flying in European airspace must now be equipped with a traffic collision avoidance system (TCAS). This system consists of a transponder by which aeroplanes that approach each other become aware of each other's presence, and of software generating an instruction to the crew what to do in order to avoid a collision. The calculation of the instruction is co-ordinated between the two approaching planes, through the exchange of signals. In this way it is ensured that the pilots get matching instructions, one of them being told to descend and the other to climb.

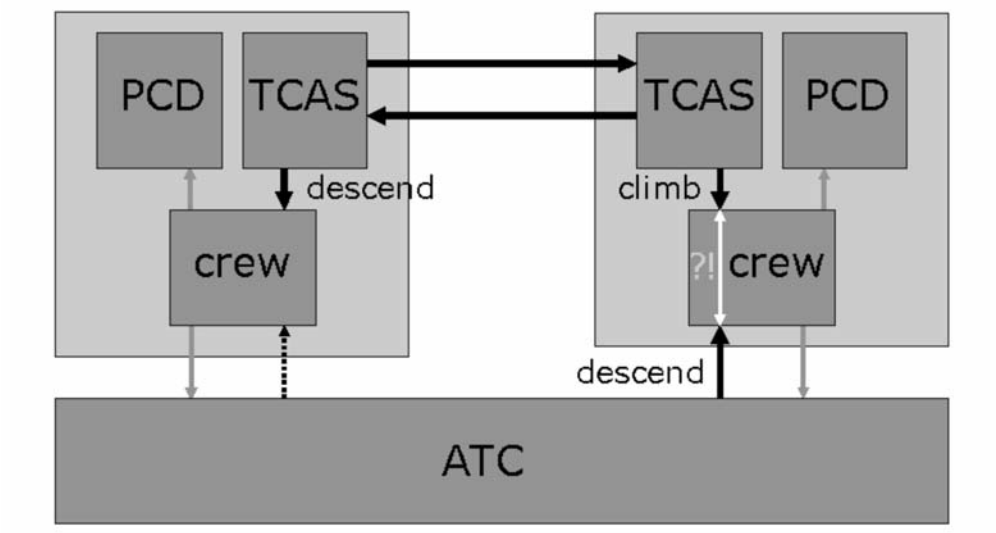


Figure 4. Schematic representation of the mid-air collision of 1 July 2002 at Überlingen, Germany.

The TCAS is a fall-back system. Normally, ATC is supposed to detect much sooner than the TCAS can become operative whether two aeroplanes are on a collision course. In the Überlingen accident, however, the air traffic controller failed to do so. He belatedly noticed the danger, but only when the TCASs of the two aircrafts were already busy exchanging their signals and generating their instructions to the crews. Just at that moment, the air-traffic controller broke in and gave just one of the two aircrafts, the Tupolev-154, an instruction to descend. This instruction, however, was the exact opposite of the instruction that the Russian pilot received just a second later from his TCAS. This led, understandably, to considerable confusion among the pilots of the Tupolev. When the air-traffic controller repeated his instruction to descend and asked for confirmation, the pilot quickly decided to follow ATC and ignore his TCAS. Since the DHL Boeing had also descended in response to an instruction from its TCAS, the only instruction it had received, the two aircraft stayed on a collision course and finally flew into each other.

TCAS was designed as an isolated system to be operative within air transport systems, and not as a subsystem of a larger air transport system. Apparently, in designing it, it was assumed that the environment in which it would be operative no longer contained, for whatever reason, an air traffic control subsystem. If the design had been undertaken from the perspective of the overall air transport system, the co-ordination of instructions and actions between ATC, aircraft crews and TCASs would have been an integral part of the design effort and it should have been recognised that when there are two subsystems present that can issue instructions to a crew, either the system should be redesigned such that it is impossible that both subsystems give instructions at the same time, or additional rules should be 'designed' that enable the crew to resolve cases where they receive conflicting instructions.

All major accidents have multiple causes, and the Überlingen accident is no exception. From a liability point of view, the failure of Swiss air traffic control agency to warn the two aeroplanes in time can be considered as the major cause. From the systems design point of view, however, it was foreseeable that such an oversight could happen and its happening should therefore not have led to a disaster. This was why the TCAS was introduced in the first place, of course. From the systems design point of view, the major mistake is rather the failure to adopt the correct system-level perspective in implementing the TCAS. This diagnosis is compatible with the engineering view of human operators: apparently there is nothing wrong with the conception of operators as mechanisms executing in an algorithmic way an exhaustive set of instructions; it is just that in this case the list of instructions was not exhaustive. And indeed, the response to the Überlingen incident reflects exactly this perspective. It has now been made mandatory to follow the instructions given by the TCAS. Once you have received an instruction, you should do as told, even when air traffic control tells you to do something different.

This, however, is exactly where the personhood of aircraft pilots becomes relevant. Being persons, aeroplane crews are endowed with responsibilities. In particular the commanding pilot carries the ultimate responsibility to ensure that no harm befalls the aircraft and its passengers. This is emphasised in almost all of the regulations that govern air transport. Responsibility, however, is a vague and fluid notion; it certainly is not an engineering concept. A key aspect of responsibility is that the person's own assessment of the situation must be central in his or her considerations of what to do. You cannot hide behind rules, you must be able to defend that the rules applied in the particular case at hand. The German bureau that investigated this particular accident discovered that there were at least ten different regulations in force which the pilots could have taken to be the one that had the highest priority, but they were inconsistent - meaning there was no general consistency on precisely this situation, where you received conflicting instructions from the hardware element of the system and from air traffic control, which is a human element of the system. From a responsibility point of view, air traffic control could be supposed to have a higher authority just for being human.

The obligation, imposed after the Überlingen accident, to always carry out the instructions given by the TCAS entails that you must also do so even when your own understanding of the situation tells you to do something different. This fits ill with the fact that the final responsibility for what happens to the aircraft remains with the pilot. It can be argued that one should never follow a particular rule or instruction blindly, in disregard of any personal assessment of the situation to the contrary. This position receives support in the TCAS case from the fact that the system is not technically foolproof. TCAS was formally proved to be correct for any situation involving two aircrafts, but it is known to be able to fail where three planes are involved. In fact a German researcher who investigated the case has suggested that the Russian crew indeed believed that there were three planes involved, due to an error in a message from air traffic control, but I do not find this convincing. However, it is known - also among aircraft crews - that the system is not correct for all situations that involve more than two aeroplanes. Such situations are not imaginary, as there are many unmanned planes or military planes in the air that are not part of this system, that do not have a TCAS installed and will therefore not be detected. So even if you draw the boundaries of the system as wide as this picture suggests, you still have elements in the air which are not included and that might be a reason for a pilot to take his responsibility.

The discussion of the Überlingen case involves exclusively the conceptualisation of operators at various levels in the system. There are also users of that system, as there must be for any engineering system, but whether one takes these to be the passengers in the plane, being flown to their destination, or the owners of the airline companies, who use it to make a profit, in either case the users are shielded off completely from the operation of the system during flight. Let me finally briefly look into a full-blown socio-technical system, where the decisions of both users and operators affect the performance of the system. My example is a motorway subsystem of the ground transport system. The users drive their cars along the motorway and the operators control the gantry indicators by which they aim to control the behaviour of the drivers and thus guarantee an optimal flow of traffic.

The difficulty here, of course, is that each individual driver is a user - that is, a human being who has a certain picture of the situation and may interpret for example speed indications as instructions or as advice or as a nuisance or as perhaps still other things. Even though by law a driver ought to interpret speed indications as instructions, we all know that few people treat laws in the same way that system operators treat the instructions that define their role. Still, this is the conception that system engineers have at their disposal. We can see every day what the consequences are for the functioning of the transport system.

In order to improve its performance, there are now attempts to apply the sort of knowledge from the social sciences that I showed you examples of previously, where people have somewhat reliable input/output characteristics. Suppose you have two routes to the same destination, a common one and an alternative one, supposedly less attractive or a lesser known one, if only slightly. Operators may try to distribute the total flow of traffic over the two routes by presenting drivers the time to reach their destination if they choose either of the two routes. If initially most cars are on the common route, an operator can tempt each individual driver to choose the alternative route by suggesting that choosing it would bring one to one's destination in a shorter time. Even if the alternative route is longer, congestion on the common route could make this plausible. Plausibility, however, need not even enter the picture; as long as a monotonic relationship exists between the number of people that change route and the indicated time difference, an operator can simply manipulate the times shown on the gantry indicators to achieve an optimal distribution of drivers over the two routes. However, since these drivers are still human beings who have a mental representation of the situation, plausibility will enter the picture. Striving for an optimal distribution of traffic may require showing highly implausible times, causing people to suspect that they are being manipulated, or to think that the signalling system has broken down, and they will no longer respond as expected.

A still more complicated situation is where you have a city with a major waterway running through it, as in London. The bridges over the river are of course crucial elements in the road traffic system. Suppose, in such a city the gantry tells you 'Castle Street Bridge blocked, turn left'. As an operator you may hope - assuming that 'turning left' is what you want drivers to do and what you correctly think is best for the total flow of traffic - that this is what drivers will do. It is quite another thing, if you must design a operator's instructions or if you must decide what message to give or must defend such a decision, whether your expectation that they will indeed turn left is justified. Drivers that know the city well, or think they know the city well, may have their own idea about whether turning left is indeed the best thing to do if Castle Street Bridge is blocked. What they think is the best thing to do will additionally depend on their expectations about what other drivers will do.

Anticipation is a major factor in traffic management, as the state of the system is constituted by the totality of the drivers on the roads: although the drivers are as users not part of the system, their cars are, and so are the drivers in their role of car operators. It is clear that it will be an extremely difficult task to design a nontrivial road transport system such that the individual behaviour of drivers adds up to a proper functioning - in terms of the distribution and flow of traffic - of the system as a whole. Modelling the system in the way that engineers are accustomed to do is hardly an option, because the theoretical and conceptual tools to do this are still lacking and may even never become available. The possibility of anticipating the actions of others is exactly where the humanity of the components of the system is most apparent, and where they most escape all attempts to measure and codify their behaviour.

These sort of difficulties are not just met by engineers involved in road transport systems. They occur everywhere where the decisions that people make - to pull a switch, to push a button - have immediate and substantial repercussions for the hardware side of the system. A typical example are the public utilities, and indeed since the liberalisation of the utilities market, the engineers involved in the operation and maintenance of the electricity network, for instance, are keenly proven engineering methods for keeping the system in working order and controlling it.

I end this paper with an overview of some conclusions.

- Socio-technical systems involve humans both in the role of operators and in the role of users. Operators are subsystems of the larger system in which they perform their operating work, and are therefore included in the system. Users are not part of the system. They are free to use the system or, in the case of a socio-technical system, to participate in using it.
- A proper functioning of socio-technical systems requires the co-ordination of the actions of all people involved, both operators and users. This will usually be accomplished through rules, and the design of such rules is therefore an integral element of the task of designing a system.

- A human decision to follow a particular rule requires first of all a judgement that the situation is one where the rule applies. But even when an operator decides that a particular rule applies, he or she can also be expected to make a judgement whether or not it is in the person's interest to follow the rule.
- The history of technology consists to a large extent in attempts to remove the 'friction' in the system that is caused by the (interpretational) freedom of operators, and many if not most of these attempts have been successful. In relation to the Überlingen accident I have seen it proposed that, rather than thinking better about the sort of instructions that operators receive, we simply liquidate the operators, in this case the pilots, completely. Operators are everywhere and continuously being replaced by completely hardware systems. This option is of course no panacea: hardware systems can fail as well, even if differently. Additionally there are institutional limits to this option, having to do with the distribution of responsibility, accountability and liability.
- Finally, regardless of the extent of automation, the friction due to the interpretational and reflective freedom of the users of the system will remain. You can never automate the users of a system, because the system exists to serve their purposes; automated users have no purposes. Although a user cannot be considered part of a system, the person who constitutes the user is present in the system in the role of operator, as is the case when an individual driver steers his or her car along the roads of a traffic network. What you can do is decouple as much as possible the user role and the operator role. The increasing interest in the development of fully automated traffic, so that ultimately the user can sleep his way from A to B, is an example of this approach.

References

1. Wilson, Brian (1990) *Systems: concepts, methodologies, and applications*, second edition, John Wiley
2. *Systems engineering: system life cycle processes* (2002), ISO/IEC 15288, first edition

7. Engineering as Synthesis - Doing right Things and Doing Things Right

Dr Chris Elliott FREng
Pitchill Consulting

Chris Elliott was an aerospace system engineer for 20 years before qualifying and practising as a barrister in environmental and public law. He now works as a freelance, helping companies solve problems where technology and the law conflict. He is also a Visiting Professor at Imperial College and the University of Bristol.

Engineering as Synthesis

I have been a Visiting Professor of the Principles of Engineering Design at Bristol for about 15 years and have consistently argued that engineering equals design. Everything else done under the label of engineering is either applied science and technology, or it is craft - making things. The element that makes engineering different from science and craft is design. That is not popular with university departments, which are mostly made up either of physicists or of people who are actually trying to pursue the craft of making things. I have the greatest respect for both groups - my first degree was in natural sciences and I have done enough craft work in my life to appreciate those who can do it properly - but my thesis remains that engineering is design.

The notes in the flyer for this seminar say that 'engineering is primarily a social rather than a technological discipline'; I cannot let that stand unchallenged. Take an aeronautical example: if you are halfway across the Atlantic, do you want to know that the diameter of the bolt that holds the engine on was calculated and the materials chosen so that it is strong enough? Or that it is there because that is the correct social context for it? Engineering is about making things that work and, if they do not work - not just in aeronautics but in many other fields, including the one in which I frequently work which is railways - people die.

On a lighter note, I have always found for almost every topic a relevant quotation from one of the leading 20th century philosophers of science, Douglas Adams. I have a line from *The Restaurant at the End of the Universe*, which is the second of the books that make up the *Hitch-Hiker's Guide* series. It concerns a party of hairdressers and management consultants who are marooned on prehistoric earth, who have formed committees to invent things to make life better. They are having a review of their work:

> 'What about this wheel thingy?', said the captain. 'It sounds a terribly interesting project.' 'Ah', said the marketing girl, 'we have a bit of difficulty there.' 'Difficulty?' exclaimed Ford, 'what do you mean, difficulty? It is the single simplest machine in the entire universe.' The marketing girl soured him with a look. 'Alright, Mr Wise Guy', she said, 'If you're so clever, you tell us what colour it should be.'

I like Douglas Adams, first because he is funny, but also because he makes many very perceptive remarks. Getting so obsessed with the 'what colour should it be?' example, when you actually miss the point about whether it goes round and carries a load, seems to be a mistaken sense of priorities. Engineering, which I repeat is about making things that work, should never lose sight of the goal.

A popular rule is that 'form follows function'. Alas not always - a great example of where function followed form was the Millennium Dome. The first decision was how big it would be in square metres, followed by the choice of material to make the roof. Only then did somebody say, 'That's pretty good - now what shall we do with it?' That is an archetypal example of letting the form dictate the function.

I once heard a speaker at a engineering dinner - and I cannot track down the source - say, 'a building designed by an engineer without the benefit of an architect is horrifying; a building designed by an architect without the benefit of an engineer is terrifying.' Let us keep a sense of proportion in engineering as a social construct.

Everyone has their own definition of engineering and mine is, 'Changing the natural world to make it better meet the needs of mankind.' Engineers are about re-forming this place from what it was originally, so that it works better to meet the needs of at least a sub-group of mankind. If you want to be biblical, this is a very complex world to design and build in six days so engineers have to finish off what God left undone. That of course invites the reaction that the history of mankind is one long snagging list.

That is getting to my central message. Engineering, in practice, is of no use unless it is sensitive to what society wants and will use. If you are stuck on prehistoric earth, any sort of wheel is worth having. However, if you are trying to design a wheel for the next generation of expensive luxury car, it will not sell if it does not look right. If it does not meet all the needs of customers for prestigious cars, it does not work. Engineering design has to be sensitive to the social context of what it designs and how it will be built.

Let me move on to the design process. When I first started lecturing at Bristol, I tried to argue that design is the art of compromise. Since I had already argued that engineering equals design, it was not long before people asked whether I meant that engineering meant compromise - at which point I managed to offend the few people I had not already offended. However, I still defend that, because there is rarely a right answer, a right design, because there are so many stakeholders who have conflicting objectives - performance, delivery time, cost, risk and many others. If the project becomes big enough to have a political dimension, you are talking about job security, national pride and international relations. There are so many axes being ground in most engineering projects and the engineer has to take them all into account.

I describe the role of the engineering designer as finding the least bad compromise that all of the stakeholders can live with. Think of it as plotting their needs on a Venn diagram and trying to find that little blob where they overlap, which everyone can live with. Of course, in almost all real engineering design challenges there is no overlapping blob. There is no common ground; the engineer has the diplomatic task of persuading someone to move his position (that is, redefine his needs) or the project is abandoned.

This is not a new concept. Shakespeare wrote in Henry IV Part 2:

> ….. When we mean to build,
> We first survey the plot, then draw the model;
> And when we see the figure of the house,
> Then must we rate the cost of the erection;
> Which if we find outweighs ability,
> What do we then but draw anew the model
> In fewer offices, or at last desist
> To build at all?

There is the principle of iteration - that you put up an idea and, if it does not fly or if the customer does not like it, you keep tweaking it and working with all of the stakeholders until you come up with something that can be done. If all else fails, you abandon it - and that is something which I suspect engineering, as a community, is very bad at. We persist, even when it does not make sense. The usual problem is that the customer wants a palace, until you tell him what it will cost, and then you start again.

It is especially true of institutional customers. Willy Messerschmitt once said, 'We can build any aircraft that the aviation ministry calls for, with any requirement satisfied. Of course, it will not fly.' This is a global problem that always arises, with the customer requiring the impossible.

The designer is left with trading off a whole load of benefits and constraints, to arrive at a compromise that everyone can work with - speed, reliability, cost, timescale, mass, comfort, the list goes on. Then there are some less obvious features, such as ease of dismantling. I have included that because I once rebuilt an old Lotus 7. I think the Chapman approach to design was built around the pop-riveter. Having assembled the mechanical parts, you then pop rivet all the panels on. Of course, you then cannot take it apart to get at the mechanics- it was never designed to be maintained, which, if you have had an old Lotus, you will realise is quite a common activity. They were always known as collectors' cars - you drive 10 miles then go back to collect the pieces! If you can't design for reliability, at least design for maintenance by making it easy to dismantle; that is part of the context of the product.

Engineering design is a mass of disciplines, not all of which are purely technical. Most projects will involve a wide range of engineering disciplines - mechanics, electrics, electronics, computing, materials … Then there is project management, including planning, construction, testing, operating and disposing. To this we must add many subjective human issues, such as biomechanics, shape, colour and form.

Because I am both an engineer and a lawyer, I tend to become involved in legal issues and particularly in health and safety, both in construction and in use. Constructors are very good at thinking about the health and safety of their workers, and users are quite good at thinking about the health and safety of their customers, but the link between the two is often missed out. Companies whose products are safe to build and safe to use are often the most profitable and successful, because safety is another one of the properties of a well-designed product that has its roots in exactly the same thinking that leads to efficiency, speed, economy and all the other desirable qualities. If the design is good, you get a whole load of consequences which - in systems engineering jargon - are called the 'ilities': sustainability, accessibility, usability, affordability and availability.

That leads into something I am very interested in, which is the ability of engineers to design something so that it still works when it does not work; what could be called partial failure or graceful degradation. I work often with railways, which are extremely complex systems - it is only once you become involved that you realise how difficult they are. Experienced railway engineers often talk about 'degraded modes', where the railway must continue to operate safely, albeit at reduced speed, when something goes wrong. Designing for degraded modes is very difficult because you are asking, how will it work when bits are not working? Engineers rarely think like that because they think their babies are perfect.

We are moving closer to the concepts of system engineering and complexity. The definition of a system that I like is, 'A system is a set of parts which, when brought together, have qualities that are not present in any of the components themselves.'

I have a little trick I sometimes do when giving talks. I take a battery, two wires with clips on and a light bulb. I clip the wires onto the battery, put it all together, and the light bulb glows. Light is not a property of any of those pieces. Light is not a property of the battery, the wires or the bulb, but it is a property of the way you put them together. That is about as simple a system as you can get, but it has an emergent property which is not obvious when you look at the pieces.

If you are going to start designing systems which have much more complex emergent properties like safety, you really have to think very deeply about all those pieces interacting, and how they interact with the people. The only way you can set about that is with an integrated approach. Every one of those myriad decisions that make up a system design have to be seen not just in the narrow context of how strong this bolt has to be, or how much current that wire has to carry, but in the much bigger context of, what is the emergent property I am trying to get and, more seriously, what is the emergent property that I am trying to avoid, like a crash?

That leads to thinking in terms of integrated system design as an overarching discipline for engineering. The Academy supports a programme of integrated system design, supporting visiting professors at a number of universities, to help the universities to prepare graduates for the real world. They should not come out saying, 'I am a mechanical engineer and that is all I do', but 'I will do the mechanical parts in the context of a bigger system.' This means kicking people out of their comfort zone. Whenever I have tried teaching this, there is a great deal of resistance from undergraduates. 'I came here to study computing', or 'I came here to study structural engineering - why are you boring me with all this electrical stuff?' Because that is how you are going to earn a living!

Future leaders in engineering will be the people who can work in a multidisciplinary team, with many branches of engineering but also all the other disciplines - and not just technical disciplines. Quite a few of my clients are public affairs companies working with clients with scientific or engineering products who have to understand and influence the political system because, otherwise - as one of them has - they find that they have been put out of business. All of those issues come up for engineers and the Academy's scheme is at least trying to expose undergraduate engineers to that way of thinking. That scheme is not dogmatic, and it does not say that you must do it this way - it is more like Mao's thousand flowers. We have produced a guidance paper, a sort of hymn sheet that the universities can tailor. It includes a little proselytising because the challenge to get some of the more dyed-in-the-wool academics to admit that there is more to engineering than a single discipline is often quite difficult. It is called '*Creating Systems that Work - Principles of Engineering Systems for the 21st Century*.' Let me quote from the Preface: 'Customers rarely want a system. What they want is a capability to fulfil a business objective. A system, be it a building, vehicle, computer, weapon or generator, is only the means to deliver the capability - housing, transport, calculations, defence or electricity. Engineers are responsible for defining, with the customer, the capability that he or she really needs, and expressing it as a system that can be built and is affordable.

The paper emphasises the importance of education. 'We must produce and employ effectively engineers who have the following qualities - creativity, analysis, judgment and leadership.' Those four have emerged with a good deal of debate. Creativity is not a passive process and you do not just follow the rules. Analysis: going back to my bolt holding the engine - this is based on hard calculation and not hand-waving. Judgment: you cannot look up answers for everything and, eventually, you have to make a value judgment. Leadership is important: if the engineer is not going to lead the project, who is? It comes back to my marketing people in the prehistoric earth, and you would not want them leading it.

That sets the scene for what we were trying to do and we have boiled it down to six principles.

1. Debate, define, revise and pursue the purpose
2. Think holistic
3. Be creative
4. Follow a disciplined procedure
5. Take account of the people
6. Manage the project and the relationships

The big message is that good design is as much about human issues as technical issues - there are three principles for each.

Finally to return to the qualities of the engineers themselves. As discussed in the first paper in this volume, people are often held to fall into one of two types, foxes or hedgehogs. Hedgehogs have one trick and they do it well - they have spikes. Foxes have many little tricks and they are cunning. The popular view is that engineers are hedgehogs and they are very good at something, while project managers are foxes, being quite good at a number of things. But designers of engineering systems have to be both; their CV is T-shaped: it has a lot of breadth and at least one deep piece - 'there is at least one thing in this project for which I am the expert'. If you cannot say that, then apart from anything else, you will not have any credibility with the others.

Those engineers have to be able to do one part of the project in detail and all of it in outline, which sets the agenda for their education. They have to know a lot of basic science and engineering -physics, chemistry and mathematics, the science on which engineering is based. They also have to have an analytical spirit - one that tries to model problems, rather than just brainstorm them. They need an awareness of the many disciplines that contribute. Finally, they need to be able to communicate with everybody - from the customer to the technician who assembles their design.

The message we are trying to convey to engineering education is, please can you think about how you form engineers who fit that pattern. This is hard to do, and it is uncomfortable for traditional engineering thinking, but it is crucial if we are to be able to engineer systems that work.

That is where I come back to my title. I started by exploring doing things right - you do not want the engine to be held on by goodwill - but, actually, doing right things is the much wider context of engineering systems.

Part III: AI and IT: Where Engineering and Philosophy Meet

8. The Engineering of Phenomenological Systems

Professor Igor Aleksander FREng
Imperial College London

Igor Aleksander is Emeritus Professor of Neural Systems Engineering at Imperial College London, having been Head of Electrical and Electronic Engineering and Pro-Rector in the College. He is a visiting researcher at Sussex University. He has researched artificial intelligence and neural modelling since 1965 and is currently contributing to research and discussions on 'machine consciousness'. He has published 13 books and over 200 papers. In 2000 he was awarded a lifetime achievement medal by the IEE for his contributions to informatics.

Phenomenology

'Phenomenology' describes rather well what I feel is an interesting approach to designing systems, and it is slightly different from the sort of things we have been used to over the last 15 or more years of AI, so that is what I am going to be talking about.

The problem with the word 'phenomenology' is that it has a spread of meanings. Primarily it is a historical movement in philosophy associated very much with the early part of the 20th century, and this is distinguished from ontology, epistemology, logic and ethics. These are all different ways of starting to do philosophy. An early contributor to phenomenology was Franz Brentano [1]. He is more often associated with the concept of intentionality, that is, having internal representations of objects that are *about* the properties of the object. Edmund Husserl [2] was probably the first phenomenologist. I have found it enormously interesting to read some of the work these people have done because it is relevant to the design of intelligent systems.

The other meaning of phenomenology is as a field of study, and it is a study of consciousness specifically based on the way that reality is perceived. It starts with what is perceived, namely reality from the first person point of view, that is, introspection, rather than what one might construe reality to be. It raises some eyebrows because in psychology, introspection was for a long time seen as a very unreliable way of proceeding. Psychologists, when doing experiments on people, need to interpret statements such as 'I am thinking about this, that and the other'. There is no way of corroborating such 'evidence'. However, if one starts designing systems with an internal point of view, one's own introspection is an interesting place to begin.

A slightly different form of words for the study of phenomenological consciousness might be: 'A study based on the way that things *seem* to be from *my* point of view which should be a decent approximation of the way things are' (non-decency would not have evolved). It seems quite obvious that if the point of view that you have of the world out there is a long way from some reality, that might not serve evolution very well. If every time you try to pick up an apple which is somewhere else from where it seems, you might eventually starve. I think there must be a reasonably close relationship between the world as it can be construed and the world as we perceive it.

Ned Block's two types of consciousness

I would like to introduce another contemporary philosopher here because he uses the word 'phenomenal' in his assessment of consciousness. To introduce Block's view I will discuss two physiological experiments which are quite fascinating. The first one has to do with blindsight. As reported by Milner, Goodale and their colleagues (1996, 'The Visual Brain in Action, Oxford University Press), there are folk who have a definite deficit in the primary visual cortex.
In other words, to all intents and purposes they are blind. But, if you throw something at them they will duck. Or if you ask them to move their hand towards a wall which has a slot in it, they will angle their hand in the right way. Ned Block as a philosopher realised that if one is talking about phenomenal consciousness one has to somehow explain this.
That is, there can be some kind of consciousness which hasn't to do with having an awareness or a phenomenology.

To add to this, Libet's experiments [3] are also very interesting. He asked people just to sit quietly and suddenly lift a finger. He was taking lots of measurements of what goes on in the brain, and he discovered that there was burst of brain activity (called the 'readiness potential'), which took place just before the person became conscious of wanting to lift their finger - this was the best part of a second or so. Here, as in blindsight, there is some activity which a person carries out and thinks they are responsible for but may be anticipated by the brain. Who is in charge?

That led Ned Block to suggest that there are two types of consciousness, and he used the word 'Phenomenal' to point to the normal situation when there is a clear inner sensation. He defined a second type of consciousness as 'Access' consciousness, which is poised for the control of behaviour. He suggests that if you have access consciousness without the phenomenal consciousness, that is the situation in blindsight and Libet's readiness potentials. But you can have the opposite as well, which is phenomenal consciousness without the access consciousness. An interesting example of this that you could be sitting in a room with a grandfather clock ticking away, and the only time you notice the ticking of the clock is when it stops.

I feel that there may be errors in the two-consciousnesses model. In neurology blindsight and readiness potential are just some of the very many brain things that go on that of which one is not conscious. Consciousness may be a result of just a very small part of the entire activity of the brain. Francis Crick and Christof Koch [4] wrote an excellent paper in *Nature* on the 'zombie' nature of some brain activity: brain activity that doesn't come into consciousness but is necessary to support brain activity that does.

A Phenomenological System

To design a phenomenological system, one has to think of a way in which such a system could be capable of having a point of view. Not only a point of view from a point in space, but a point of view of itself in the world that it perceives. If we build a machine that has this point of view, then it can use its point of view and the way it perceives itself in the world to act on the world, an attribute that might be of considerable use in the design of a robot.

Critiques of GOFAI

Going back to Good Old-Fashioned AI, it is worth recalling that many of the criticisms were based on a lack of phenomenology or intentionality in computational representations. Dreyfus, who wrote several books on what machines can't do, was himself a phenomenologist. Weizenbaum believed that any form of intelligence - he didn't actually talk about consciousness - in a rule-based system was the result of an attribution by the user rather than something the machine was doing; and John Searle's celebrated Chinese Room thought experiment seeks to show that symbol manipulation is not the same thing as meaning manipulation or an understanding of the world out there. [5]

A Visual Comprehension Test

Here is a visual version of the Chinese room argument.

Figure 1

Take the picture above. I could ask a question such as where do you think that is? The notice, if one could read it, might indicate that French is spoken. You might notice that to enter the door you have to be careful not to trip up on the wooden platform; you don't have to worry about the dog because it looks quite docile, and so on. The point I am making here is that the classical AI way of dealing with a comprehension test with a picture like that would be to use a semantic network such as shown below:

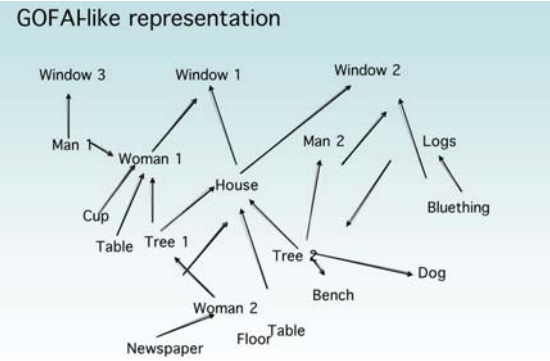


Figure 2

However, it's quite impossible to use a semantic network unless you can work out ahead of time the myriad of questions that are likely to be asked of the system. The simple point I am making here is that there is a kind of deferral of representation in living systems. In other words, I am not worried about making semantic networks in my head about what I see over there. As No_ and O'Regan point out [6] it's enough to have a complex world out there and then act on it, get engaged with that world, that seems to be very important. Of course, if one then has an imagination of that world, this has to be at a fine level of grain to act in lieu of the world itself.

We stress that phenomenology is not scene analysis. It implies that perception comes from an active engagement with the world that is being perceived or its imaginational high-grain representation of a sense of self in the 'out there' world and what that self can do.

The Machine Consciousness Paradigm

I would like to talk for a little about the machine consciousness paradigm and how phenomenological systems fit into that. Some of us were speculating about machine consciousness in the early 1990s, and then in 2001 there was a meeting at Cold Spring Harbor Laboratory organised by Christof Koch (neuroscientist), Owen Holland (robot engineer) Rod Goodman (information engineer), and David Chalmers (philosopher). They got about 20 or so people together, a mixture of philosophers, engineers (computer engineers mainly) and neurologists, to ask the question of whether it does make sense to talk in terms of "artificial consciousness" or "machine consciousness" in the way that one talked about artificial intelligence or machine intelligence.

This is Christof Koch's concluding comment in the final report: "The only near universal consensus of the workshop", it is worth noting here because not many people agreed with each other about what consciousness was, but the one thing they did agree about, that whatever it was: "in principle one day computers or robots could be conscious. In other words, that we know of no fundamental law or principle operating in this universe that forbids the existence of subjective feelings" - which is a commonality of the various definitions of consciousness - "in artefacts designed or evolved by humans".

So this is still an open challenge. Of course someone could say, let's define consciousness as that which can't be done by machines. That would be one definition that cannot be satisfied, but most other sensible definitions of consciousness go towards saying that machines could have subjective feelings.

The paradigm has developed since then and there are many conferences now on machine consciousness. The field does divide between non-phenomenological approaches and phenomenological approaches. I will go through some examples.

Bernard Baars' Global Workspace

In the middle 1990s Bernie Baars, as a psychologist working at Gerald Edelman's Neurosciences Institute in California, described an early and influential model of how consciousness might work in a mechanistic way [7]. I will leave that for the moment and come back to it. Stan Franklin of the University of Memphis decided to build a machine called IDA, which stands for Intelligent Distribution Agent (Journal of Consciousness Studies, vol 4-5, 2003). It was funded by the US Navy to billet sailors via email. Sailors were conventionally billeted by email by human billeters who were finding new jobs for these sailors, asking them questions as to what their preferences were, whether they liked hot climates, cold climates, etc. which enabled the billeters to attempt to find a suitable post. When tests were done with the intelligent distribution agent, people were reporting that the billeter had become more caring and sympathetic than was the case

previously. It is at that level that Franklin decided that in order to do machine consciousness one has to build AI systems that in some way capture various emotional concepts. It is because the system seemed to handle emotions and feelings pretty much like (according to Baars) a human being that the system was brought for discussion to the Cold Spring Harbor Laboratory meeting.

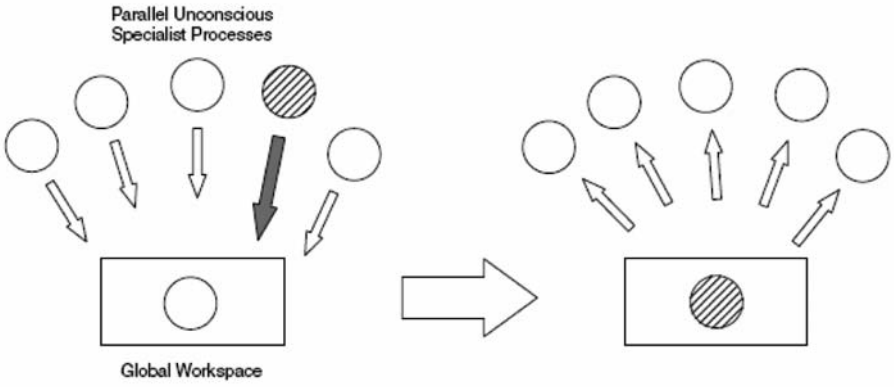


Figure 3. The global workspace architecture

Taken from M.P. Shanahan

Murray Shanahan at Imperial College who has provided a very simple explanation of how global workspace works (see figure above). There are many processes competing to get into the global workspace. One will win, mainly on the basis of some form of saliency, some form of relevance (it could be a memory process) to whatever is happening to the organism at that particular time. Once it enters the global workspace it is broadcast back to all the other (unconscious) processes. It is the broadcasting of the winner of a whole lot of competing processes which Baars has defined as being the conscious activity at that particular time. Because this is a dynamic system that runs continuously, and the continual broadcast time after time of the system is similar in some way to what William James would call the stream of consciousness. Shanahan has used this in an interesting way. He has a global workspace-type digital arrangement where parts of the system have a self-rehearsal possibility and then a competition for entering the global workspace, and when something enters a global workspace it influences the processes again. Shanahan has used world representations in these competing systems. In other words, the system tends toward the phenomenological because it begins to work on the basis of what the world seems to be like in those processes.

Owen Holland: first holder of an engineering grant for Machine Consciousness

Psychologist and engineer Owen Holland was the first researcher to have received a grant for investigating machine consciousness in robots. Cronos, a machine which Holland built has an important physical structure which is akin to a human skeleton with its compliant characteristics

Holland's computational philosophy starts with a model of the world and a thinker in it, but the thinker is capable of thinking about the world and, in thinking about the world, it is capable of recognising that the thinker itself is in the world, so this is a phenomenological step. But then the thinker is also capable of distinguishing between himself and the world and working out that each has a different sphere of influence. In the overlap between these spheres of influence, one can build plans for an interaction of the object with the world, which gets the robot to go on and do things.

Designing Phenomenological Conscious Machines

In trying to design machines which in some way have phenomenology as the core of the system, there is a set of requirements that appear to be paramount. To be phenomenological, a machine S must contain machinery that represents in explicit fine grain what the world and S within it seem like from the point of view of S. The insistence on fine grain follows from the example of the visual comprehension example given above and provides the necessary deferral.

To accept the above sequence of requirements, a key factor that will then be common to all machines that support phenomenology is something which I call *depiction*, because the word *representation* is not right. Traditionally representation refers to symbols and depiction is about as far from symbols as you can get. A depiction is a state or part of a physical brain that is as close a reflection of the world as the sensory apparatus permits. This points back to what I've said about representations as having to have some sort of decent relationship with the world otherwise the object that uses them is going to go wrong.

I would like to make quite clear this is not early Wittgenstein-like 'pictures in one's head' because there is so much more to a depiction than just a picture. In fact, it does not have to be topological. Consider the following diagram:

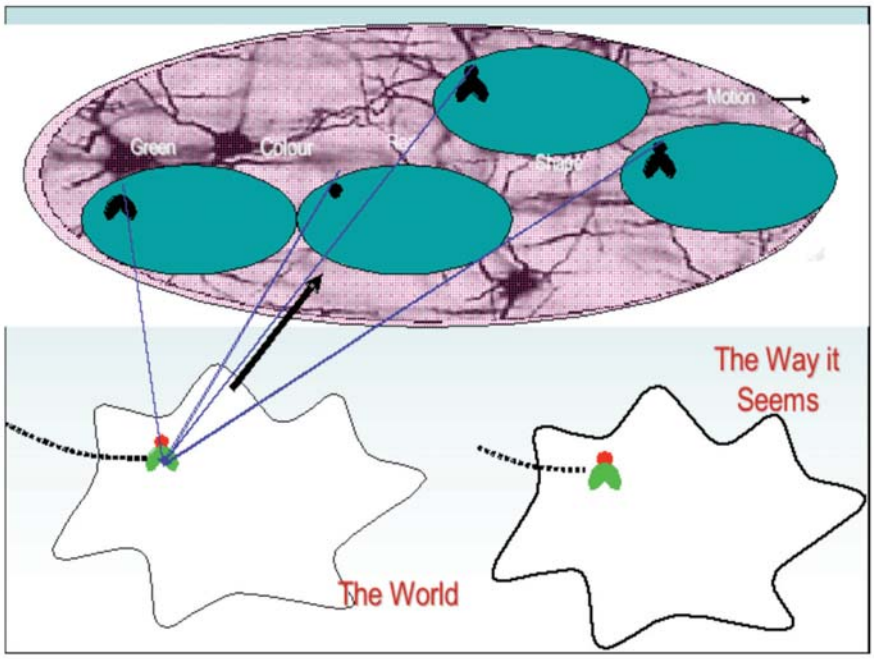


Figure 4

Here is the world in which a little red-and-green fly has landed. There is a human observer or a primate observer looking at this, and the first thing we know about the visual system is that, whatever has happened in the world just gets torn apart and thrown about all over the brain, in the early part of the visual system in particular. There is an area for colour, an area for green things, an area for red things, an area for shape and an area for motion. Semir Zeki has written a beautiful book on the visual system [8], which highlights a major question for neurologists: how do these distributed representations come to us as one coherent vision? This is called the integration problem. In broad terms, I have suggested that the elements of the world scene are indexed by muscular signals of where in the world the element may be found. This is the depiction mentioned earlier.

Simple Visual Depiction

Here is a very simple example of depiction. Imagine seeing a little black fly on a white screen jumping from the left half to the right half of the screen. The involvement of muscles in providing a proper depiction of this event is shown simply below.

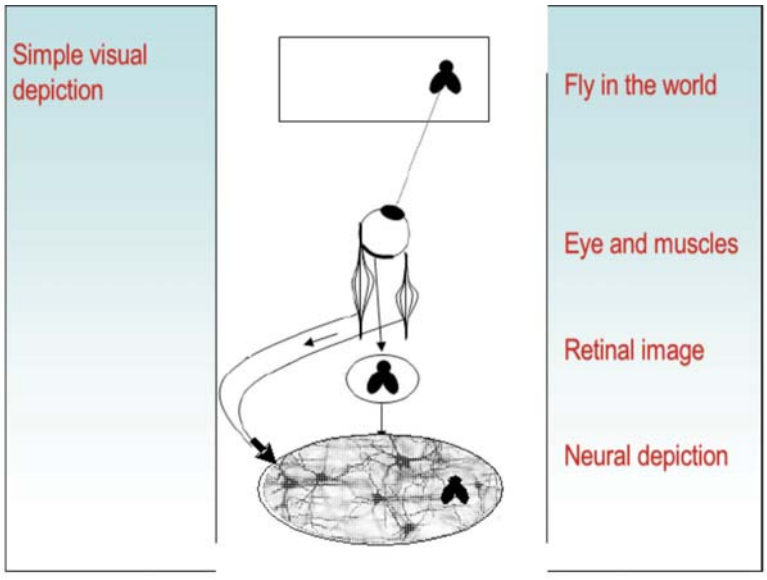


Figure 5

The picture at the back of the retina remains the same whether the fly is on the left or the right, but what makes a big difference is the musculature that moves the eyes around, and that has to be incorporated into the description, into the depiction in some way. That is a very simple way of looking at it.

What Makes a Depiction?

If we want to have a fuller view of depiction it involves almost everything. It involves eye movement, neck movement, even arm movement when you point at something, leg movement and tactile information. All of this comes together in order to create a good representation of what the world might be like.

So the answer to what makes a depiction is the influence of motor actions on sensory neural representations which need not be topographic. There is a vast corpus of literature that shows that the brain has many cells in its sensory processing systems are indexed by musculature and therefore decode where elements of sensation are in the out-there world.

What mechanisms for Synthetic Phenomenology?

I am very briefly going to mention the fact that in working with consciousness I cannot possibly think of it as just one thing, so I have divided it into several components, each of which makes it easier to find some underlying machinery. I list these below and indicate that an explanation may be found in my book [9].

1. Presence in an out-there world
   Depiction through involvement of musculature
2. Imagination, experienced and constructed
   Memory through recursion in neural nets and language influence
3. Attention (Exogenous and Endogenous)
   Selection mechanisms in 1. and 2.
4. Volition
   Using 2 to find what is possible
5. Emotion
   Evaluating what's possible to find what is wanted

It has been possible to create a 'kernel' architecture that incorporates the above five 'axioms'.

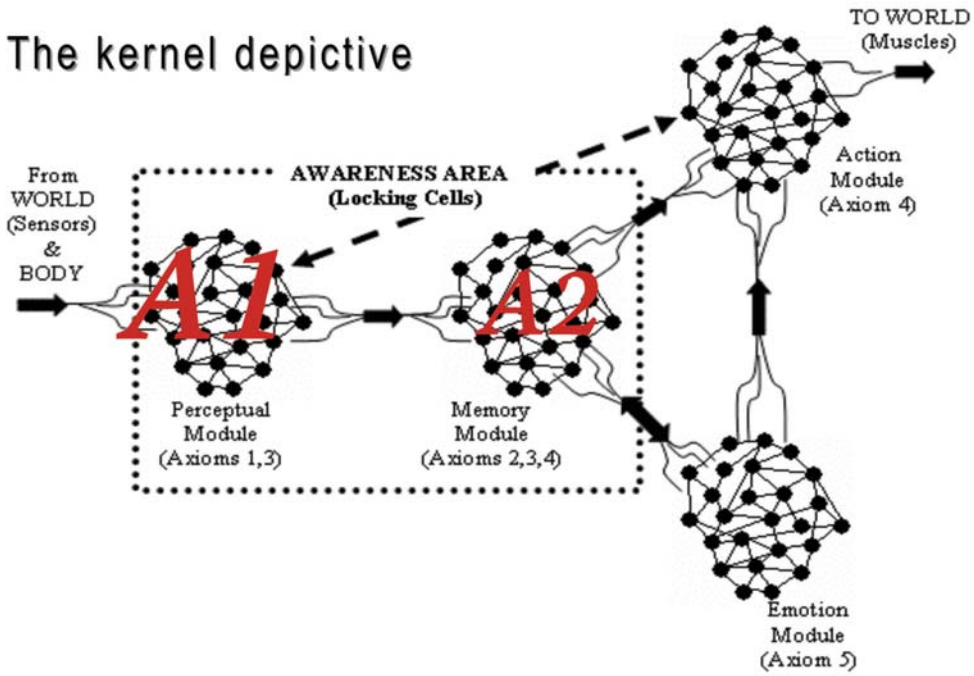


Figure 6

The four groups indicate neural network implementation with A1 and A2 in this structure as the main components that support consciousness in the depictive sense discussed. The other two areas provide action and emotional input to the awareness areas, but do not directly enter the conscious sensation.

When what "seems" is not what is

To conclude this presentation I have taken an example where what seems is not what is as in the well known Necker cube shown below.

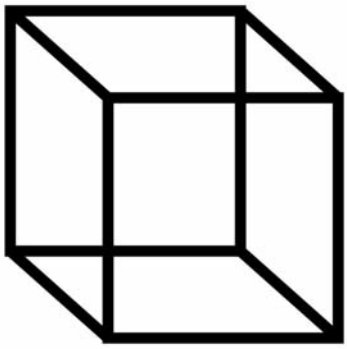


Figure 7

Here there is just a heap of lines on the page, while many see this as a cube where sometimes one face comes forward and sometimes it goes to the back reversing regularly while it is being watched. An explanation of this has been attempted for well over 100 years. A vast number of papers have been written about it. Suggestions include: is it eye fixation? Experiments show that this is not the case. Is it rapid habituation of the visual pathways? They can't habituate that fast. Richard Gregory has an AI model that it is an ambiguous memory access. In other words, you put in an ambiguous address to a memory and the filing cabinet sometimes responds with one folder and it sometimes responds with the other. There is no such thing as a computer-like memory access structure in the brain.

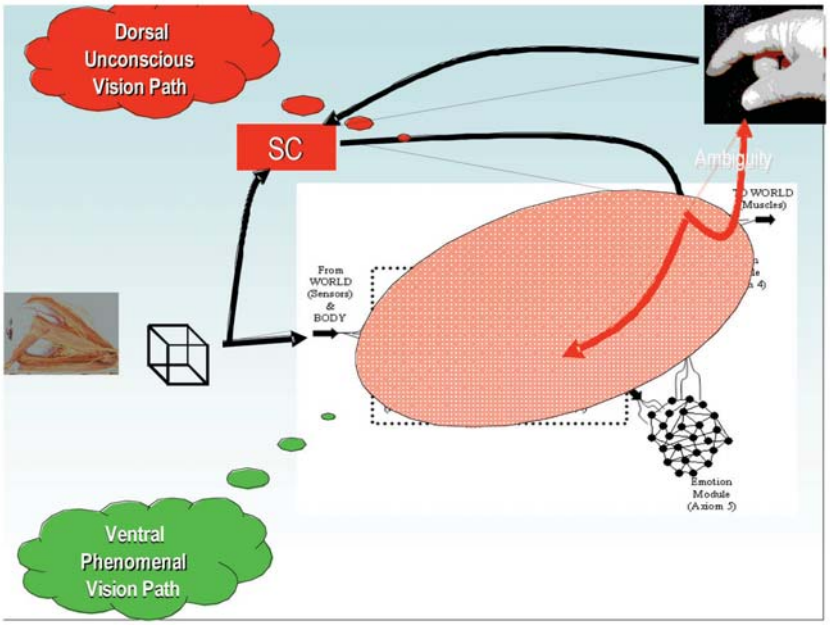


Figure 8

This is a model we have built which comes back to the blindsight question. There are two paths that emanate from the early visual part of the brain. One (the ventral) is the path that helps us see, gets into the visual cortex and so on. The other is a bypass that goes directly to the motor areas (dorsal). So this is how some blind persons may be capable of avoiding an object or positioning their hands in line with a slot. They are not aware of what they see but they can create actions from visual input even though their phenomenological vision doesn't work. Our hypothesis is that an ambiguity due to the Necker cube can arise in the dorsal path. SC stands for superior colliculus; which primarily causes eye movement, and also receives input from the motor areas which may be trying to set up the fingers to grab the cube. The ambiguity in this 'grasping' loop may cause an instability which feeds back from the unconscious motor areas to the conscious ventral visual areas. This is what O'Regan calls the "grabbiness" of something that we see.

The above structure, a neuro-representation modeller, has areas that are unconscious - all the pinky stuff - it has areas A1 and A2, and it is a composite of those two that come into awareness. I will just show you this working. [In the semianr digital model was demonstrated to show (through its phenomenological display) that this hypothesis is plausible.]

A simple concluding thought

When the technology starts hitting a dead-end, as it did with AI a few years ago, why not try glancing at the philosophy to obtain new engineering ideas.

References

1. Brentano, Franz (1874,1995) *Psychology from an Empirical Standpoint*, Trans: Rancurello et al. Routledge (Original in German)
2. Husserl, Edmund (1913, 1963) *Ideas: A General Introduction to Pure Phenomenology*, Trans. Boyce Gibson, Collier
3. Libet, B. (1978) 'Neuronal vs. subjective timing for a conscious sensory experience' in *Cerebral Correlates of Conscious Experience* (Buser, P. A. & Rogeul-Buser, A., eds.) pp 69-82. Amsterdam: Elsevier/North-Holland Biomedical Press,
4. Crick, Francis and Koch, Christof (2001) 'The Zombie Within Us', *Nature,* 411, 893,
5. See Boden, Margaret (2006) *Mind as Machine*, Oxford University Press, for an authoritative account of the history of AI
6. No_, J. Kevin and O'Regan, Alva (2001) 'A Sensorimotor Account of Vision and Visual Consciousness' *Behavioral and Brain Sciences*, 24(5), 939-1011
7. Baars, Bernard J. (1997) *In the Theater of Consciousness: The Workspace of the Mind*, Oxford University Press
8. Shanahan, M.P. (2006) 'A Cognitive Architecture that Combines Internal Simulation with a Global Workspace', *Consciousness and Cognition*, vol. 15 pages 433-4498.
9. Zeki, Semir (1993) *A vision of the Brain*, Oxford: Blackwell
10. Aleksander, Igor (2005) *The World in My Mind, My Mind in the World*, Exeter: Imprint Academic

9. Interactive Empiricism: The Philosopher in the Machine

Ron Chrisley
COGS/Department of Informatics, University of Sussex

Dr Ron Chrisley is the Director of COGS, the Centre for Research in Cognitive Science at the University of Sussex, where he holds a Readership in Philosophy in the Department of Informatics. He has held various research positions in Artificial Intelligence, including a Leverhulme Research Fellowship at the University of Birmingham and a Fulbright Scholarship at the Helsinki University of Technology, as well as positions at NASA-Ames, Xerox PARC, the Stanford Knowledge Systems Laboratory and ATR Laboratories near Kyoto. For the past fifteen years he has also been an occasional visiting lecturer and researcher at the University of Skövde. He was awarded his doctorate by the University of Oxford in 1997, and is the editor of Artificial Intelligence: Critical Concepts (Routledge 2000).

I would like to discuss some thoughts I've had recently about the general question we have been considering, that of the relationship between philosophy and engineering, and to present some possible new directions for collaboration between the two fields.

Take-home message

The conclusion I am heading for is this: I think there can be a two-way beneficial interaction between philosophy and engineering. Two qualifications can be made at the outset. First, I am sure there are possible collaborations other than the particular two-way interaction I'll be looking at. Second, a proper evaluation of the proposed interaction would include an investigation into the history of engineering to see if one can find examples of this two-way interaction, both to illustrate the interaction, and as a kind of validation of the fruitfulness of the interaction. I have not yet conducted such an investigation. But even if there have not yet been any interactions between philosophy and engineering of the form I have in mind, I maintain that the possibility is worth consideration, particularly because such interactions may be necessary for some philosophical breakthroughs and engineering achievements.

One of the more contentious claims I make to allow room for the mode of interaction I have in mind is this: some philosophical breakthroughs can only come about (or at least they are much more likely to occur) if philosophers, or the people who are struggling with the relevant conceptual questions engage in engineering; that is, design, build and interact with working systems appropriately related to the questions being considered. This is contentious in that most philosophers seek to draw a sharp line between a *priori* and *a posteriori* enquiry, with the discipline of philosophy entirely on the former side of the divide. On such a view, there is no room in philosophical methodology for scientific inquiry, even if it reliably yields knowledge of the world: the truths it discovers are empirical, whereas the truths philosophy seeks are not, nor can they be established by consideration of such. However, the results of scientific enquiry (empirical truths) at least have the same format (propositional form) as the results of a priori inquiry, the universally accepted *modus operandi* of the true philosopher. All the more irrelevant, then, does engineering seem to philosophy. Not only does it traffic in the contingent and particular rather than in the necessary and universal, but it also fails to be a mode of enquiry at all, in the sense of a process that yields propositional knowledge.

I propose that this view is a misunderstanding of what philosophy is, or at least what it can be. It is only quite recently that such a sharp distinction has been made between philosophical and empirical forms of enquiry. I suspect many philosophers of the past, possibly including Kant, and definitely including Vico, would be sympathetic to the idea that engaging in engineering can lead to philosophical advances. Perhaps they had insights we would do well to recover.

There is another way that philosophy and engineering can interact, that is operates in a direction opposite to the interaction just mentioned; not by engineering helping philosophy, but by theorists/philosophers themselves being components that contribute to the dynamics of working systems. It may seem a bit odd, but what I will propose is that for the case of some complex systems, for instance an artificial consciousness, it might be necessary to incorporate a theorist or philosopher into the design. That might not make much sense right now, but later I will illustrate how this is possible by giving an example from actual work in artificial intelligence where I think this is a good way of understanding what is going on.

Thus, the incorporation of a philosopher or theorist into the design of a system has two aspects. One can consider the effect that the system dynamics will have on the theorist or philosopher. One can also consider the effect that the theorist/philosopher has on the artefact's dynamics; I will give a concrete example of that from some work at MIT.

Direction 1: Engineering conceptual change

The first direction, that of making philosophical progress by doing engineering, has to do with conceptual change.

One way of understanding philosophy is that it is about trying to solve conceptual problems. That this is not always appreciated by everyone is at the heart of a joke I was told long ago by Brian Cantwell Smith. Some people, he said, make fun of philosophers for having struggled with very simple questions for millennia without having come up with an answer. For instance, consider the old chestnut: if a tree falls in a forest and no one is around to hear, does it make a noise? Some criticise philosophers for still pondering that question after all this time. But on the other hand if you give that question to scientists, they'll scratch their heads for a while, go away, write some things down on paper and then come back and say, 'Well, we've worked it out for elm and birch but we're still trying to solve the general case.'
The relevance of the joke in this context is that the kind of answer the scientists gave is a sign that they didn't really understand the question; they have mistaken a conceptual problem for an empirical problem.

Philosophy is about trying to resolve these conceptual problems. The orthodox way to solve such problems is through conceptual analysis, a *priori* enquiry, as discussed before. Nothing that I am going to say implies that such enquiry should not continue; I join the vast majority of philosophers in my conviction that not all limitations on our understanding, even scientific understanding, of the world are merely a matter of not having enough data. In particular, our problems in trying to understand consciousness are conceptual; the obstacles we face in understanding what it is for a physical thing to also be an experiencing thing aren't just a matter of not having enough knowledge of the nervous system.

Even if we knew much more about the nervous system than we do now, we would still have some fundamental puzzling questions. We have a naturalist intuition that consciousness, like anything else, is at root physical. Indeed, I am assuming here a broadly physicalist perspective: the belief that every kind of phenomenon in the world is grounded in physical happenings. On the other hand, we have another intuition - the philosopher Dan Dennett calls it the "zombic hunch" - that it is possible for there to be a creature - a zombie - that is physically just like us, but "there's no one home": it isn't conscious. At least some people believe that it is not inconceivable that there could be someone who is physically (and thus behaviourally) identical to you and yet different from you with respect to its experiential properties, even to the point of not having any experiences at all: a zombie-you. Those two intuitions are in direct conflict: the naturalist intuition is that like everything else, consciousness must be, at root, a physical phenomenon, whereas the zombic hunch implies that even if you fix in the physical you still haven't fixed the experiential. The presence of both of these intuitions produces an unsatisfactory cognitive dissonance.

One way of responding to this is to diagnose the cognitive dissonance as the result of flaw in our concept of consciousness. If our concept of consciousness has paradoxical implications, perhaps we should try to develop a new concept of consciousness that doesn't. Perhaps we should look to conceptual change as a way to resolve this problem of the conflict between our naturalistic inclination and the zombic hunch.

Conceptual conceptual change?

The suggestion that we solve our conceptual problems by changing our concepts is rather facile; it immediately prompts the question: how can we do this? One constraint is this: we don't want to change the subject. We want to change our concept of consciousness in a way that ensures that in employing the new concept, it is consciousness that we are still talking and thinking about; it is just that we are doing so in a better way. When we think about gold using our concept of gold, we think about the same thing that the ancients thought about when they thought about gold using their concept of gold. But we are not employing the same concept they did. We have a better concept of gold; we know what the essence of gold is - having an atomic number of 76. Although the ancients were confused and had many false beliefs about gold, it didn't mean they weren't thinking about gold. In fact, we can only make sense of their beliefs being false after we first understand them as being about gold.

Can we do the same thing for consciousness; can we refine our concept of consciousness? I propose that we can, and that we need to do so in order to solve some of the conceptual problems we face. But it seems unlikely that the kind of conceptual change required can itself come about through merely conceptual processes. By conceptual processes, I mean such processes as adding propositions to, or subtracting propositions from, one's stock of beliefs, whether it be by learning some more facts about consciousness or about the brain, or by engaging in philosophical arguments. Also, creating a new concept out of logical combinations of the concepts one already possesses. Such methods seem unable to surmount the impasse we have reached. If the recent history of discussions and disputes in the areas of consciousness studies and the philosophy of consciousness is any indication, there is no rational way to convince

somebody who has the zombic hunch to not have the zombic hunch, and vice versa [1]. I am sceptical that the kind of conceptual change required can be achieved simply by reading journal articles about consciousness and other purely linguistic, propositional modes of inquiry. Don't get me wrong: of course these modes are essential to developing our understanding of anything, including consciousness. But it seems to me that there is reason to believe that they are not enough to resolve certain intractable conceptual problems, especially in the case of consciousness.

Non-conceptual conceptual change

If we are going to change our concept of consciousness so that we can make scientific sense of consciousness and understand the place of consciousness in the natural world, we might have to have our concept of consciousness undergo what you might call a non-conceptual change, a non-conceptual development of our concept of consciousness. What do I mean by non-conceptual development of a concept? I mean changes to a concept that aren't simply a case of adding or subtracting propositions to one's stock of beliefs, nor a case of creating a new concept out of logical combinations of the concepts one already possesses. But I am not concerned with just *any* kind of non-conceptual change to one's concept. Getting hit on the head, for instance, might change what you think or what you think you think about consciousness, or undergoing neurosurgery, or perhaps taking certain kinds of psychoactive drugs. Perhaps you will have an 'aha' experience if you do some of those things; perhaps some of them will result in non-conceptual conceptual change. But these are not, you may be happy to hear, the kinds of non-conceptual change I have in mind. The methods of change I do have in mind, like the bad examples just given, can't be achieved by hearing a philosophical argument, or by reading a passage of text. But unlike those bad examples, the changes are still *rational* changes that are *justified*, and in particular are *based on the experience of the subject matter*. Unlike the bad examples, where the change in your conceptual state is non-conceptual but random and not justified in any way, I am looking for ways that we can change our concept and yet have it be a change that tracks reality in some way, even if it can't be summarised in some text, or even if it can't be transmitted through text. This might sound impossible, but I think it is actually commonplace.

Concepts as skills

To see why this might be so, we need to get a clearer view of what concepts are. Consider the famous duck-rabbit figure, or the Necker cube. Wittgenstein argued that what underlies being able to move between the different ways of seeing the Necker cube or the duck-rabbit is the 'mastery of a technique' ([2] p 208). This is exactly the kind of ability we are looking for in the case of conceptual change: the capacity both to see something in a new way and to see how it is the very same thing as what you saw in the old way. To be able to see how a thing that is appreciable from the experiential perspective is also the very same thing that is understandable from the physical perspective, to move seamlessly between those two viewpoints, is a skill. If so, then we now know that what we need in order to resolve the conceptual problems of consciousness is a skill: Acquiring the right concept of consciousness is a matter of acquiring a skill.

Note that skills are usually such that they can't transmitted through text alone. I can't give you a piece of text and thereby give you the ability to ride a bicycle. We can't have a philosophical argument that will give you that skill. Rather, skill in a domain typically requires experience of that domain. For instance, some symbolic, linguistic, propositional, information (the advice of a friend) helped me learn to juggle. But it wasn't sufficient for me to acquire that skill. It's true that I tried to acquire the skill without that conceptual knowledge and didn't do very well; it was only when my friend helped me, through language, to draw attention to certain aspects of my experience, I was able to juggle, in my own feeble way. But the advice alone didn't give me the skill; I had to attempt to juggle in order for the advice to be of any use. So also for the case of understanding consciousness. I am not saying there is no role for argumentation, thinking, reading journal articles, etc. I am only saying that these are probably not enough, and we need something else; we need a skill that cannot be acquired conceptually.

Interactive empiricism

The ideas that skill acquisition is a way to achieve non-conceptual conceptual change is an important part of a view that I call "interactive empiricism". This is distinct from empiricism *simpliciter*, that all concepts must be grounded in sense experience. Interactive empiricism is not a species of empiricism in that sense. Rather, it is the view that the possession of *some* concepts requires having a particular kind of sense experience, a kind usually not emphasized in traditional empiricism. The "interactive" in "interactive empiricism" indicates what this particular kind of sense experience is: the sense experience involved in *interacting* with the subject matter of the concept, the stuff the concept is of.
To have the kind of concept that will solve our conceptual problems, one must master a technique of understanding how one's perspective on the subject matter - in this case consciousness itself - will change in the light of one's different

ways of intervening in that subject matter. Both the application and acquisition of this skill rerquire the having of experiences in the context of interaction with the subject matter of consciousness. Riding a bicycle isn't merely a passive reception of input of the kind the traditional empiricists were thinking of when they talked about grounding ideas in experience. The experience is interactive, the result of a dynamic engagement with the world. One acquires the skill of riding a bicycle because one experiences sensory feedback *in response to* one's actions; the kind of experience goes beyond a mere a one-way input from the world to your ideas.

To make that a little more clear, I want to draw attention to the fact that at least some movements in cognitive science are finding that interaction is essential to understanding cognition. Interaction is essential to perception on some views, such as O'Regan and Noë's sensory-motor contingency theory [3]. On that account you can only be perceiving the world if you have some capacity to interact with it, or if you are actually interacting with it. Consciousness in general is thought to involve interaction on views such as Susan Hurley's; hence the title of her book, *Consciousness in Action* [4]. Cognition in general is thought to involve interaction essentially on some views such as Mark Bickhard's; hence the title of his 'Interactivist Manifesto' [5]. A nice illustration of a concrete way in which interaction is crucial to certain kinds of cognitive development, in this case in visual perception, is the classic study by Held and Hein [6]. They placed neonate kittens with undeveloped visual systems in an apparatus consisting of a circular room with a bar suspended from the ceiling, able to rotate about its midpoint. At the end of each bar is a harness for a kitten. The harnesses are such that one kitten is touching the ground and is able to move around relatively freely, but the other kitten is suspended in a way that its movements will not change its position in the world at all; rather, its position is determined by the movements of the other kitten, which is able to walk around more or less normally. For the first kitten, there is a very natural interdependence between its actions and the visual input it receives. For the second kitten, there is little or no correlation between its muscle movements and the visual input it receives, because the visual input is largely determined by the first kitten. No matter what the second kitten does with its limbs, it doesn't, in the main, affect the input it receives. The result of this study is striking; after the developmental period; the first kitten has more or less normal vision, while the second kitten is more or less blind. This shows that having the right kind of interaction, engaging in action and having sense experience that is appropriately related to those actions, is crucial to certain kinds of development.

Meta cognitive science: Theorist as subject

Perhaps our conceptual development shares this property with visual development in kittens. If this is a general cognitive principle that governs our conceptual development as well, then we some developments in our concepts, for instance our concept of consciousness, may also require us to intervene in a subject matter and then receive reciprocal experiences that are appropriately related to those interventions. In the case we are considering, the interventions will in the phenomenon of consciousness itself.

A general science of human cognition should apply to individual cognizers; specifically, it should apply to cognitive scientists, philosophers, and engineers. And if one's cognitive science says that cognition in general, and conceptual development in particular, is interactive, it may also be that making philosophical advances via conceptual development will necessarily involve engaged, experiential activity.

Engineering as interaction

This is where engineering comes in. The kind of interaction that is relevant to understanding consciousness, cognitive systems, artificial consciousness, et cetera, is the design and construction of actual working systems that model or exhibit consciousness-related phenomena. It seems to me that these kinds of interactions will be the kind that will allow this conceptual development, that allow us to acquire the skills that constitute a conceptual advance with respect to consciousness.

I don't need to be so restrictive here; I don't need to deny that there might be other kinds of interaction that could assist in conceptual development with respect to consciousness. For instance, in interacting with each other or interacting with subjects in the experimental laboratory, if we were not only interacting in a more or less normal way, but also had access to real-time scans of each other's brains, this might also be a way of having the kind of interactive experience required to develop our concept of consciousness. But this is a much less plausible idea than the engineering based approach that I am suggesting here. First, there are the ethical issues: true interaction would require intervention on the lowest, neural level: directly altering another's neural state. But there are non-ethical problems as well. Compare trying to understand, say, how a computer works in a similar way, that is while you are interacting with the computer you have an oscilloscope and you can see what is going on in the lowest level hardware level of the computer while you are typing into Microsoft Word or something. In theory maybe you could get some great insights,

and acquire some skill that would constitute a better concept of computation, but it seems unlikely; it is just too much of a jump from the lowest to the highest level to expect you to see some interesting correlations. There needs to be a step by step, level by level, structured approach that will allow interaction to have an effect on our concept of consciousness. So also, then, with the brain scan suggestion. By contrast, designing and building cognitive systems can and does have this mediated structuring of activity, and is thus more likely to be the kind of interaction that is going to yield the right kind of conceptual change.

(An aside for those familiar with Jackson's Knowledge Argument [7]: Once one realises that science is itself an interactive activity that involves experiencing the world, the whole premise of Jackson's argument is revealed to be contradictory. Jackson asks us to imagine that Mary knows everything science has to say about colour vision but has never seen red. On the view of science presented here, this is revealed to be a contradiction. As Alter [8] has independently observed, Jackson assumes that all the knowledge of physical science can be written down and acquired by Mary through reading. If what I have said here is right, then this assumption of Jackson's is false. Science in general involves the having of experiences, and colour science in particular involves the having of colour experiences. So for Mary to truly know everything that science knows about colour vision, she would have to have all the experiences that scientists have needed to have about colour vision, in particular the experience of seeing red. To suppose this and to also suppose that Mary has not had the experience of seeing red is a contradiction.)

Direction 2: We are a part of the systems we build

To close, I would like to say a few things about the other direction of collaboration between philosophy and engineering. The key observation here is that we are part of the systems we build, and just as interaction can have a salutary effect on the philosopher, as just discussed, so also it could be that the philosopher or the theorist might be a crucial component in the developmental dynamics of that system. Not only is this an abstract possibility; it is also a concrete actuality in that some research in robotics is exploiting this means of interaction. To introduce this research, let me ask: What is the biggest engineering advance in artificial intelligence in the last twenty years? A provocative answer is: Kismet's eyebrows [9]. Putting eyebrows on the robot named Kismet may very well be one of the biggest technological and conceptual advances in artificial intelligence in the last twenty years. Let me explain. Kismet needed to learn how to track visual objects. It could only do that if the stimuli were appropriate; that is if they were moving within a certain range of speeds at a certain distance so Kismet could focus on them, etc. One could try to ensure that the trainer kept the stimuli within this range by a number of means. But a very efficient way of getting the trainer, a person, to keep the system within a particular part of the phase space was to put eyebrows on Kismet, and maybe adding a little more like having Kismet jump back and raise its eyebrows under certain conditions. When the state variable moved out the optimal region of phase space, Kismet jumped back and raised its eyebrows. Given our inbuilt dispositions to respond to such situations, one doesn't have to tell the trainer what to do in such a situation; one doesn't have to give them instructions, they don't have to consult a rulebook or anything. The trainer will just respond naturally, because we are built to respond to displays of "startlement" in particular ways, and the trainer will be non-conceptually disposed to treat raised eyebrows and pulling back as a display of startlement. The fact that Kismet is in fact not an experiencing creature, and therefore unable to be startled, is irrelevant. What is relevant is that Kismet behaves like an experiencing, startled creature, and therefore the trainer will respond unreflectively in the appropriate way. That is, the trainer will pull back, will move the stimulus back into a proper part of the phase space with the result that tracking and learning will continue. That is a very efficient way to exploit the dynamical relationship between Kismet and the trainer in order to get Kismet to learn in the proper way. That is an example of a way in which the trainer, the theorist, the philosopher even, can be in the loop, be part of the system.

Combining the directions of interaction

These two directions of interaction can be combined. If we are part of the system, not only can the theorist/philosopher have a beneficial causal effect on the robot's performance, as we saw with Kismet; but, as we saw in the first part of the talk, interactions with the robot can also have a beneficial effect on the theorist/philopher by prompting conceptual change. This suggests an alternative design strategy for artificial consciousness. Instead of trying to design a machine consciousness in one step, we could instead see the design as a dynamical developmental process. On such a view, the first step is to design a system S1 in such a way that it will prompt relevant conceptual changes in us, such that those conceptual changes will allow us to design another system S2, so that S2 will prompt further conceptual changes in us, and so on. On this view, we see ourselves in a dynamical, a dialectical relationship with the systems we build and try to get on that trajectory, get onto that design spiral, rather than trying to get to the endpoint all in one go. If we think about this development, this design trajectory and apply some of the techniques of engineering to that trajectory, perhaps we will be able to get further in the quest for machine consciousness than we have been able to so far.

Frank Herbert was a prescient author; he wrote about this possibility in the 1960s in a novel entitled *Destination: Void* [10]. In the novel, people attempt to create machine consciousness in an indirect way. First, they genetically engineer clones to have the right sort of skills and personalities to form a team which might make inventing machine consciousness more likely. These clones are then put into a carefully engineered technological environment, which included certain kinds of computing technologies and neural wetware, but which was located on a spaceship. Then these clones are manipulated and given certain kinds of motivations; specifically, they discover that the ship they are on is going to fail if they didn't create a machine consciousness first. The hope of the designers of the whole clone/spaceship/ hardware system is that if those ingredients are put together in the right way, the clones will come up with a design for a machine consciousness, or at least come up with the next stage of such a design, which can be the starting point for the next generation of clones.

Thus, Herbert already anticipated the idea that including the engineer/theorist/philosopher into the design of the system might be essential for the construction of machine consciousness. Another interesting point is that a crucial part of the project in the novel is that the challenges the clone crew face are such that they are forced to think about what they mean by the word "consciousness". The engineered crises force the crew to engage in philosophy, and attempt to come up with a definition of consciousness. The crises are such that the crew can only see what consciousness is by being confronted with the embodied exigencies of the crises. The people designing this experiment don't know what consciousness is, but are rather hoping that they have assembled a conjunction of crew, situation and technology that will allow advance toward a solution to be made; "designing for emergence".

Although Herbert's work is mere science fiction novel, perhaps it isn't so far off from what we are or could be doing now. I don't just mean Kismet, although that is a good example, and I praised it as being a substantive breakthrough in artificial intelligence. I also mean research into creative technologies, environments that facilitate creative processes, document systems that facilitate creative insight, and even devices that induce brainwave patterns believed to be correlated with creative activity. These kinds of technologies, if they were applied to the particular case of developing machine consciousness, would be ways of pursuing the bi-directional mode of philosophy/engineering design I have been discussing.

Finally, my own work on the SEER-3 project and "synthetic phenomenology" [11] is another example of an application of this design strategy. This research aims to produce a robotic system such that the understanding gained by interacting with it permits the specification of experiential states that are not easily specifiable by non-robotic, non-interactive means. Thus SEER-3 is another example of how the skills one acquires by interacting with one's own artefacts might be useful in making some progress on understanding consciousness.

References

1. Dennett, D. (2005) *Sweet Dreams: Philosophical Obstacles to a Science of Consciousness*, Cambridge: MIT Press
2. Wittgenstein, L. (1972) *Philosophical Investigations*, Oxford: Blackwell
3. O'Regan, K. & Noë, A. (2001) 'A sensorimotor account of vision and visual consciousness', Behavioral and Brain Sciences 24(5): 883-917
4. Hurley, S. (1998) *Consciousness in Action*, Cambridge: Harvard University Press
5. Bickhard, M. (2008) 'Interactivism: A Manifesto', forthcoming in Campbell, R.L., Ó Nualláin, S., & Bickhard, M.H. (Eds.), *The Study of Mind: Toward Inter- and Intra-Disciplinary Cooperation*. Available at **www.lehigh.edu/~mhb0/InteractivismManifesto.pdf**; accessed 11/11/07.
6. Held, R. and Hein, A. (1963) 'Movement-produced stimulation in the development of visually guided behavior', *Journal of Comparative and Physiological Psychology* 56(5):872-876.
7. Jackson, F. (1982) 'Epiphenomenal Qualia', *Philosophical Quarterly* 32:127-36
8. Alter, T. (1998) 'A Limited Defence of the Knowledge Argument', *Philosophical Studies* 90, 35-56
9. Breazeal, C. and Scassellati, B. (2000) 'Infant-like Social Interactions Between a Robot and a Human Caretaker', *Adaptive Behavior* 8:1
10. Herbert, F. (1966) *Destination: Void*, Penguin.
11. Chrisley, R. and Parthemore, J. (2007) 'Synthetic Phenomenology: Exploiting Embodiment to Specify the Non-Conceptual Content of Visual Experience', *Journal of Consciousness Studies* 14(7):44-58.

The Royal Academy of Engineering

As Britain's national academy for engineering, we bring together the country's most eminent engineers from all disciplines to promote excellence in the science, art and practice of engineering. Our strategic priorities are to enhance the UK's engineering capabilities, to celebrate excellence and inspire the next generation, and to lead debate by guiding informed thinking and influencing public policy.

The Academy's work programmes are driven by three strategic priorities, each of which provides a key contribution to a strong and vibrant engineering sector and to the health and wealth of society.

Enhancing national capabilities

As a priority, we encourage, support and facilitate links between academia and industry. Through targeted national and international programmes, we enhance – and reflect abroad – the UK's performance in the application of science, technology transfer, and the promotion and exploitation of innovation. We support high quality engineering research, encourage an interdisciplinary ethos, facilitate international exchange and provide a means of determining and disseminating best practice. In particular, our activities focus on complex and multidisciplinary areas of rapid development.

Recognising excellence and inspiring the next generation

Excellence breeds excellence. We celebrate engineering excellence and use it to inspire, support and challenge tomorrow's engineering leaders. We focus our initiatives to develop excellence and, through creative and collaborative activity, we demonstrate to the young, and those who influence them, the relevance of engineering to society.

Leading debate

Using the leadership and expertise of our Fellowship, we guide informed thinking, influence public policy making, provide a forum for the mutual exchange of ideas, and pursue effective engagement with society on matters within our competence. The Academy advocates progressive, forward-looking solutions based on impartial advice and quality foundations, and works to enhance appreciation of the positive role of engineering and its contribution to the economic strength of the nation.

The Royal Academy of Engineering promotes excellence in the science, art and practice of engineering.

Registered charity number 293074

The Royal Academy of Engineering
3 Carlton House Terrace, London SW1Y 5DG

Tel: 020 7766 0600

Fax: 020 7930 1549