Chapter 1.

INTRODUCTION

At the turn of the millennium the annual spend on ICT (information and communication technology) was of the order of $1,500 billion. Does the world obtain $1,500 billion worth of benefits from this spend? According to studies made over the past twenty years 1, the failure rate of IS has consistently been well over 50% [Footnote 1]. So why do we -- why does humanity -- pour $1,500 bn every year into ICT? Would that money not be better spent on health, overcoming poverty, or reducing climate change and environmental damage?

Such considerations challenge us to seek to understand what is going on and why the benefit is so much less than the amount we are so willing to spend.

# Perhaps it is because our technologies are not yet good enough?
# Perhaps it is because we do not use ICT in the right way or for the right things?
# Perhaps it is because IS development is inefficient?
# Perhaps we too readily accept the way ICT controls our lives, and the assumption that ICT is the solution to everyting?
# Perhaps our very assumptions about the nature of ICT and what it can do for us are mistaken?

The purpose of this book is to introduce a different way of looking at ICT and information systems (IS), and to suggest some tools to help us do this. The tools derive from philosophy but are orientated to everyday experience of ICT and IS. And those tools will be applied to five areas of research and practice in ICT/IS to which the above questions relate.

Walsham [2001,p.251] says his whole professional interest may be encapsulated in the question "Are we making a better world with IT?" But then, on the next page, he asks "What is meant by a better world?" The different way of seeing ICT/IS introduced in this book aims to provide the basis for answering both of those questions.

But wait! Why do we need a different way of looking at ICT/IS? An alternative question may be raised:

# Perhaps the very posing of the question above is itself to be questioned. After all, what might we mean by $1,500 bn worth of benefit?

The usual way to argue that a different way of thinking is needed is first to show there are problems which the old ways of thinking cannot solve and then show that the new way can solve them. Under such a strategy the first half of this book would be devoted to examining each of the extant ways of understanding ICT/IS to argue conclusively that they cannot solve the problem. But such a strategy has two weaknesses.

One is demonstrated by the alternative question itself. The very posing of a problem depend on, and comes from, a particular way of looking at things. The types of reasons one must marshall to argue that we need a different way of thinking, are not objective facts, pre-given, but are themselves generated by what is on offer as ways of thinking. Our setting out of the overall problem above presupposes we can understand ICT in terms of financial costs and benefits. The very statement of the problem does not make sense unless we think in those terms. This is not to say there might after all be no problem, but it underlines that our very way of looking at things dictates how we see them as problematic. So the fact that someone can uncover a problem only shows that s/he has a particular way of looking at it, not that a new way is to sought. In such a broad area as ICT/IS no truly conclusive argument is likely to be made that a different way of understanding is actually necessary.

The second weakness is that most arguments for a new way of looking at IS/ICT presuppose they must demolish previous ways to make way for the new way. That is not what this book wants to do. It does not want to take an adversarial stance. It might indeed be able to replace existing ways of thinking, but it can also, instead, engage with, support, augment and enrich existing ways of looking at IS and ICT rather than replace them. So, what is presented in this work is not so much a new way (connotation: replacement) of understanding IS/ICT as a different way (connotation: enrichment).

This book does not adopt the usual strategy. Instead, it presents a proposal to be considered, tried, applied, refined and tested -- and perhaps eventually rejected -- rather than an argument that existing ways must be replaced. The book does not have a substantial section devoted to demolition. It begins construction from the start. During the process of construction -- that is, of developing and exploring the different way of understanding ICT/IS - indications will be given of where it might have advantages over existing ways of thinking. Some of these indications become meaningful because of the different approach itself, though most are issues that have already been recognised and commented upon by researchers and practitioners in the various fields of IS or ICT. Some readers might then use it to formulate new frameworks to replace extant ones, but others might use it to support, critique and enrich them.

Now let us return to phenomenon of IS/ICT with which we began. The title of the book is ’Philosophical Frameworks for Understanding Information Systems’. Most of the rest of this chapter is devoted to clarifying what is meant by ’philosophical’, ’frameworks’, ’understanding’ and ’information systems’. They will be examined in reverse order.

1.1 INFORMATION SYSTEMS: WHAT ARE WE TRYING TO UNDERSTAND?

"The field of computing appears to be in a state of transition." said Glass, Ramesh and Vessey [2004] "Although historically it has evolved as several stovepipes of knowledge -- predominantly ... CS, SE and IS {computer science, software engineering and information systems} -- there is now some impetus for amalgamation." In an analysis of research and thinking in these three fields, they noted that an increasing number of 'integrated schools' of computing are being formed.

Yet other 'stovepipes of knowledge' have emerged, distinct from their three, such as the information society [Lyon, 1988] and what Landauer [1996] called 'the trouble with computers': the benefits or detrimental impact of information technology used as part of human living and working. Those working in artificial intelligence (AI) often differentiate themselves from computer science, because their central interest is not algorithms and data structures as such but the nature of computers and especially in what way they resemble human beings. However it began to be clear during the 1980s that, while traditional software engineering techniques might be suited to the research laboratories and to the generation of well-defined software for well-structured applications, a radically different approach was needed to cope with the ill-structured applications in business and such approaches as Soft Systems Methodology [Checkland, 1981] has become popular.

1.1.1 Areas of Research and Practice is IS, ICT

Each such 'stovepipe of knowledge' constitutes a distinct area of research and practice, in which a different set of issues is meaningful, and a different set of assumptions is made about what is important, what is deemed a 'problem' that needs solving, and what is good practice and research.

To take just two examples, those working in computer science, for example developing better graphics-handling facilities, are concerned with algorithms while those trying to gain benefit from employing computers in business, for insurance brokers using an electronic placement system [Walsham, 2001], are concerned with the benefits or detrimental impact the system has on themselves as insurance brokers and on their customers.

Neither is of much interest to the other. Those who practise in each area see things in different ways and are driven by different motivations. Research methods in each area are different -- those of computer science, logic and psychology in one, those of sociology and economics in the other. Different types of theory are generated and different conceptual frameworks assumed.

Should they be of interest to each other? It seems unreasonable to demand that insurance brokers, their customers or other stakeholders (as they are called by the research community that studies these things) should concern themselves with graphics algorithms (even assuming the IS they are using displays its information graphically). It unrealistic to expect those who design graphics algorithms to think of all possible beneficial and detrimental impacts from use of all possible programs that might conceivably use their algorithms.

And yet there remains a thin thread of responsibility here in each direction, which implies that the two areas of research and practice that concern themselves with use of IS and algorithm design are not to be seen as entirely independent of each other. Which graphics algorithms are available as raw materials to IS developers will influence the shape of the IS they develop, which will in turn affect its use, and conversely the requirements of users will feed back to influence the design of future raw materials.

Over the last few decades there has occurred a differentiation of areas from each other. In each area, debate arises about what may be deemed appropriate research criteria and methods for the area, and what issues or problems are within the remit of the area, within its horizon of meaning. There has been a working-out of the rich diversity and coherence of each area. Researchers and practitioners in one area share largely the same set of concepts, issues and concerns -- and these differ markedly from those in other areas. As this occurs, each area become conscious of itself and discussion ensues on whether each constitutes a discipline, and how to define it.

The delineation of areas cannot, therefore, be an exercise in seeking complete independence between them. Nor is it one of differentiating theory from practice, because in each area there is both theory and practice. For example, while it might be tempting to see computer science as 'theory' being employed in business 'practice', there is practice as well as theory in computer science and there is theory as well as practice in business use of IT.

Each of the vignettes in the preface indicates a distinct area of research and practice. There are doubtless more, but the following areas relate to the vignettes:

# the shaping of basic technological resources (in the vignette's case, knowledge representation languages and tools),
# the beneficial (or detrimental) usage of computer artefacts or systems in human life,
# development of such artefacts or systems,
# the nature of computers and information (including the artificial intelligence question of whether computers can think),
# ICT as a kind of technological ecology, in which we live and which shapes our very way of lives, but which humanity itself shapes.

These five areas may be differentiated by the relationship humanity has with the technology, respectively as its shaper, its user, its developer, its conceiver, and its denizens. The five questions posed at the start are expressions of these five areas, though in a different order.

1.1.2 Problems and Issues in IS, ICT

Each area exhibits different types of problem and different 'lifeworld' (that is, everyday, pre-theoretical) issues, all of which contribute to the overall problem and lifeworld of ICT. Some examples include:

# Shaping of technological resources: inappropriate basic resources, unnatural or difficult to use by IS developers. Everyday issue: the diversity of types of meaning to be represented.
# Usage: a system might be beneficial in one way to certain stakeholders but detrimental in other ways to others, for example the placement system mentioned above did not support Utmost Good Faith, which is central to placement. Everyday issue: the complexity of impacts, especially that many are indirect, unanticipated or long-term.
# IS development: Development of the wrong system, for example failure to recognise an important aspect of use (such as Utmost Good Faith). Everyday issue: the dynamics of the development team, such as loyalty to the project.
# Nature of computers: The artificial intelligence question of whether computers can truly understand, which exercised much discourse up to the 1980s remains unresolved, and degenerated into dogmatic positions. Everyday issue: how computer present themselves as multi-layered things to us as we use or develop them.
# From the curse that is email which, though it promised to make communication quicker and easier, consumes increasing amounts of our time to impact of globalization on Africa. Everyday issue: How then shall we still live in an ICT-dominated world?

It is these five areas for which philosophical frameworks for understanding will be formulated in this work, though in a different order. Other areas doubtless exist, and it is hoped that the approaches exhibited in this work will provide readers with the capability to formulate other frameworks for understanding other areas.

Though such differentiation and delineation of areas continues, as Glass et. al. have indicated there is also a move towards integration. More generally, a desire is emerging to recognise what we might call the 'whole story that is information systems', in which each area plays its part, for example in the work of the late Enid Mumford [Avison, et. al., 2006]. The discussion in this book recognises this'whole story'. The challenge to integration is that what is meaningful in one area frameworks is often completely meaningless in others. For example, there is no room in logic programming (a computer-science framework for understanding how to represent knowledge) for the normative issue of differentiating between beneficial and detrimental impacts of IS in use.

What both researchers and practitioners need is to have some way of understanding, or looking at, the area(s) in which they work. But what is ’understanding’?

1.2 UNDERSTANDING

The range of issues that are meaningful in an area is vast. For example, issues meaningful to the use of computers include: human computer interaction, user interfaces, ergonomics, user-computer dialogue, proximal use of tools, use by individuals to accomplish their tasks, use by organisations to reap benefit, playing computer games (such as MUDs) in which the ’user’ engages with virtual characters and items, immersion and presence in virtual reality, social relations around the computer, impacts of using IS and how it comes about, unexpected and indirect impacts, diversity of impacts such as communicational, social, economic, legal or ethical, multiple stakeholders, benefit versus detrimental impact, success versus failure, use of ICT to change social structures, and to challenge assumptions, and much more.

What does it mean to understand these? Researchers and practitioners need a basis for understanding not only each of such issues as are relevant to them, but also the relations between them, and also how to expand their understanding as new issues arrive. How may we -- how can we -- obtain a way of understanding all potentially relevant issues of an area of research and practice in a coherent way?

1.2.1 Problems with Theoretical Frameworks

A time-honoured way is to construct theories that explains some of the issues, and/or methodologies that guide them. For example, ergonomics, Davis’ [1986] Technology Acceptance Model (TAM) and Latour’s [1987] Actor-Network Theory (ANT) each raise and address important issues. But such ways of understanding usage focus on a narrow range of issues. This is not a deficiency in them, because most do not claim to do more than that, but it is a danger. The danger is that that very focus can lead researchers and practitioners to assume that nothing else is meaningful about IS use, and so other issues become downplayed, suppressed and ignored. For example, ergonomics downplays the higher levels of HCI, and completely ignores issues of benefit in use. TAM is much wider, purporting to cover both ease of use and usefulness, but it tends to do so in a managerial way. ANT claims to escape this, and especially to focus on both micro and macro levels, but can downplay the normative difference between benefit and detriment. The same tendency to divert attention away from some important issues may be found with theoretical frameworks in any area.

Information security is not an area that is covered in this book; that is left to another occasion, when a framework for understanding security may be developed by following the methods sketched here. But security experts Bruce Schneier and Niels Ferguson understand the problem of narrow approaches. As they say in the preface of their book Practical Cryptography [2003],

> "Arguing about whether a key should be 112 bits or 128 bits long is rather like pounding a huge stake into the ground and hoping the attacker runs right into it. You can argue whether the stake should a mile or a mile-and-a-half high, but the attacker is simply going to walk around the stake. Security is a broad stockage: its the things around the cryptography that make the cryptography effective."

'Understanding' is not restricted to theoretical knowledge, but is orientated to practice or, as Midgley [2000] calls it, 'systemic intervention'. This requires a 'broad-stockade' approach.

In development of IS to supports business tasks the knowledge that must be elicited is multi-faceted (Jacob and Ebrahimpur, 2001:78):

> "The majority of the managers have a long history in the company and successful leadership of projects is dependent on long tenure since the preference is for managers with a generalist competence profile. In the words of one interviewee: One needs to have as broad a knowledge base as possible. It is the outer parameters that one must have knowledge about."

Though loosely defined, 'outer parameters' indicates facets that are often overlooked. In interdisciplinary applications of IS 'outer parameters' abound.

The 'everyday' of both practice and research is rich and full of surprises. If frameworks are to help us understand any area of information systems, then they must be able to cope with the diversity of the everyday and be open to surprises as they arrive. Theories and theoretical models can provide insight into specific issues, but they should never be allowed to divert attention away from other important issues. For this reason, for example, cost-benefit analysis is grossly insufficient as way of understanding IS usage.

1.2.2 Lifeworld-Oriented Frameworks for Understanding

The type of understanding sought in this work is that of what philosophers have called the lifeworld. This is the everyday life of both practice and research in each area, a stock of shared understanding that has a background character. While much of our understanding is theoretical and involves explicit conceptual structures, some understanding is pre-theoretical, or a-theoretical, an understanding that is intuitive, cultural, embodied in aspiration and attitude, much of which cannot be made fully explicit. Philosophers call this the 'naïve' attitude (without any negative connotation). This book seeks to respect the importance of all such things in formulating frameworks for understanding, and to formulate frameworks that will be sensitive to such things in the everyday experience of practice and research in each area. As far as this author is aware, such an approach to understanding is not common in the five areas delineated above.

The words 'lifeworld', 'everyday', 'pre-theoretical' and 'naïve' will be used almost interchangeably.

There are two ways in which the lifeworld can be used as empirical input to inform our attempts to understand an area: we may be guided by the content of everyday experience, or its structure. The content is the actual beliefs and assumptions people hold, and it varies with times and cultures: the lifeworld content of mediaeval Europe was very different from that of today. But the structure of the lifeworld is its nature and characteristics regardless of culture and content. Sometimes it might not be clear whether something is structure or content, but in most cases, in this book it is the structure that must guide us rather than the content. Therefore, it is useful to make explicit some the characteristics of such a lifeworld, everyday approach to understanding each area of research and practice.

Husserl first used the term ’life-world’ to differentiate it from the ’worlds’ of sciences like physics, psychology, sociology, and following Husserl’s lead, many thinkers have reflected upon the lifeworld and highlighted other characteristics. Some of the more important characteristics are as follows, and some implications for lifeworld-oriented frameworks for understanding (LOFFU) are mentioned. They are very much summarised here, but are expanded later.

The lifeworld as basis for intersubjectivity. Husserl argued that the sciences cannot operate without a stock shared life-experiences which give meaning to the concepts used in science. This implies a LOFFU should be open to such stocks of shared experiences within the field it tries to understand.

"The life-world is above all the province of practice, of action" [Schutz and Luckmann, 1973,p.18]. This implies a LOFFU should concern itself with practice as well as research.

The lifeworld is diverse. Schutz and Luckmann, 1973,p.20] give an example of the diversity of types of sub-actions in writing a letter from wielding a pen, making marks, to asking a friend; they spoke of ’provinces of meaning’, transition between which requires a ’leap’. This implies a LOFFU should be open to accommodate diversity of meaning. Many extant frameworks are too reductionist in character for this.

The lifeworld is a "pre-given reality with which we must cope" [Schutz and Luckmann, 1989,p.1]. Likewise, Husserl called our attention "to the things themselves". This givenness extends to types as well as individuals [Schutz and Luckmann, 1973,p.229]. This implies a LOFFU should enable the researcher and practitioner to ’listen’ to the world of their area as it presents itself to them, without trying to force it into a priori conceptual structures they bring to it. Subjectivist ways of looking at IS/ICT can have difficulty here.

Lifeworld is engagement. Heidegger, though he did not use the term ’lifeworld’, stressed that in everyday life we engage with things rather than distance ourselves from them as ’rational’ thinkers or actors. This implies a LOFFU should not assume that either researchers or practitioners in its area are ’rational actors’ who always deliberately plan and rationally deduce, taking a stance ’above’ the area, but that they are closely engaged with it. Many extant frameworks do assume the rational actor.

The lifeworld resists being made explicit. It has a background character (Husserl). The lifeworld with which we engage we are hardly aware of and take for granted (Heidegger). The lifeworld is that curious thing that dissolves as we try to take it up piece by piece (Habermas, cited in Honneth, Knodler-Bunte and Windmann [1981:16]). This implies a LOFFU should not assume that researchers and especially practitioners in the area always do, or even can, express what is relevant or important explicitly. LOFFUs should always be suspicious of explicit statements. Positivist frameworks can fail here.

The lifeworld exhibits meaning and normativity. Habermas, based on Weber, stressed the normativity and meaning that we experience in the lifeworld. By contrast ’systems’ (such as the economic system of money or the political system of voting) work by mechanical rules that are supposedly devoid of meaning and normativity. This implies a LOFFU should have place within it for meaning and normativity, as well as structure and process. Many extant frameworks do not.

The lifeworld attitude differs fundamentally from the theoretical attitude. Several thinkers have stressed this (Schutz called it a ’natural attitude’). A theoretical attitude observes or acts at a distance, tries to formulate an explicit theory or method, into which it tries to squeeze all that is experienced, and evaluates or justifies things by reference to that theory or method. An everyday, natural, lifeworld attitude can understand things more intuitively, can evaluate or justify by reference to common sense (including norms), and does not try to reduce the diversity experienced to a theory or method. A LOFFU should act in the latter way rather than the former.

However, as explained later, the lifeworld attitude need not be uncritical. It can incorporate theories and methods. Giving due respect to ’the things as they present themselves’ does not entail naïve realism, nor a purely interpretive stance. In everyday life we often adopt a critical stance to something, including to social structures and assumptions.

The above insights into what constitutes the lifeworld will be taken further once the philosophy adopted in this work has been introduced, at the end of chapter 3.

1.3 FRAMEWORKS

A framework for understanding (FFU) an area is a way of seeing the area. It can guide both research and practice in the area and itself might emerge out of and be refined by such research and practice over the years. Some frameworks for understanding an area are explicitly stated while others might be tacit. Some have arisen from practice, others from research. Some explicit frameworks arising from research appeal to philosophy.

1.3.1 Frameworks in Each Area

In each of the five areas mentioned earlier (but in a different order) a number of frameworks for understanding the area have arisen. (The references given may be found in chapters 4 to 8 respectively.)

Human use of computers may be seen as:

# dialogue with a distal agent,
# proximal use of a tool [Winograd and Flores, 1986],
# psychological behaviour,
# a matter of cost versus benefits,
# networked activity (Actor-Network Theory),
# power, conflict and emancipation.

Nature of computers and information may be understood as:

# electronic hardware,
# programmed bit manipulation machine; information is the bit,
# stored symbols manipulated; information is symbols,
# agent; information is knowledge,
# all these at different levels [Newell, 1982],
# like human beings (artificial intelligence),
# a new reality of bodiless mind (cyberspace).

Information systems development (ISD) may be seen as:

# programming,
# structured development (e,g, Waterfall Model),
# iterative development,
# knowledge elicitation and knowledge representation,
# "orchestrating changes of appreciation through a cyclic learning process" [Bergvall-Kåreborn, 2001].

Shaping of the technological building blocks (algorithms, computer languages, etc.) used by IS developers may be seen in the following terms:

# Developers can performs wonders with: first-order predicate logic (logic programming), or relations (relational data model), objects and classes (object-orientation), and so on.
# They need a complete KR ontology (e.g. Wand and Weber).
# We should aim for ’KR to the people’ [Brachman, 1990].
# Alexander’s Design Patterns are the way to go.

The information society or technological ecology may be addressed in terms of:

# Technological determinism,
# Social shaping of technology,
# Giddens’ [1993] Structuration Theory,
# ’Revealment’ (Heidegger),
# the ’silicon idol’ [Shallis, 1984] or a ’religion’ [Noble, 1997],
# a ’liberating vision for technology’ [Schuurman, 1980].

Some of these ways of seeing things in their area have been made explicit, thought out and given philosophical basis, while others have not been linked with philosophy, not been fully thought out, or not even been made explicit. But the latter are, nevertheless, frameworks for understanding because those who research and practise in an area share a set of beliefs, concepts, norms, etc. that are meaningful.

Frameworks within an area often disagree with each other over some methodological or philosophical point, and indeed many have arisen by means of a paradigmatic reaction against earlier ones. But frameworks from different areas are more fundamentally incommensurable with each other because they operate in completely different horizons of meaning.

1.3.2 Issues that Constitute Frameworks for Understanding

The framework for understanding adopted by us influences or determines the conceptual apparatus we employ when working in those areas -- how we classify things, what theories we devise and kinds of methodologies and rules we formulate to guide our practice, which scientific paradigm guides our research, but also what we see as important in the area, what types of questions we find ourselves asking, what we see as problematic and what we allow as possible solutions. (This is why, earlier, it was argued that it is not appropriate to try, in this book, to prove conclusively that a different way of looking at IT and IS is necessary.) A FFU cannot be proven either correct or incorrect by theoretical means because it is a set of beliefs and assumptions about the area that is adopted or adoptable by those working in the area, held as a pre-theoretical commitment.

To formulate a framework for understanding an area explicit involves surfacing a number of things assumed by those working in the area in both research and practice. Those who research in an area know which research methods are appropriate, by what criteria to judge the quality of research, and how individual research problems relate to others in the area. They also assume a conceptual structure [Meister, 1999] and what issues are of current importance. Scientific paradigms, as discussed by Kuhn [1970], may be part of a FFU.

But those who practise in an area need, in addition to these: an awareness of the normativity of the area, which defines what is deemed to be ’problem’ and what drives aspirations of those working in the area, and methodology of the area by which ’solution’ may be achieved. Behind any area flows a history, at its root lies a motivation, and at its centre stands an assumption of the position human beings take in relation to the area and other presuppositions about the nature of reality. FFUs must have some place for such all things as well as for the more obvious things.

To discuss and justify research and/or practice in an area fully, and how the area relates to others, we must take all such things into account and have a basis on which they may be understood. A good FFU will reach to the horizon of meaning of the area. Such things are addressed by philosophy.

1.3.4 Characteristics of Frameworks

The kind of framework needed for the purposes of this work may be characterized as follows.

# A framework. Not necessarily formal, nor fully expressed in written form, nor even fully explicit. But its main themes or principles are made explicit in a reasonably systematic way. It involves not just concepts related together but also attitudes, commitments, beliefs, aspirations and cultures.

# Open. A FFU should be open to extension, but in a way that is true to its nature, rather than by merely bolting new pieces on.

# Coherent rather than logical. The elements and characteristics of a FFU should fit together comfortably even though it might be impossible to present an unassailable logic to justify it. For example, the framework developed in chapter 4 for computer use tries to do justice to both the structure and the normativity of various human-computer relations. Logic alone would be content with doing justice to structure without normativity. But the very notion of ’do justice to’ only makes sense if we do justice to all that is relevant; to do justice to structure but not normativity is inconsistent with the very notion of justice.

# Guiding. The frameworks should provide a basis for guiding both research and practice. It need not include a methodology as such but should enable methodologies to be developed by researchers and practitioners and critically evaluated. For example, Soft Systems Thinking is a framework while Soft Systems Methodology [Checkland, 1981] is a methodology therein.

# Whole Story. The frameworks should be able to link, not by accident but self-consciously, to other frameworks, whether these are alternative frameworks for the same area or frameworks from other areas. The former is necessary in order that the frameworks developed here should enrich and support rather than replace existing ones. The latter is necessary if we are to have any chance of being able to address the ’whole story that is IS’.

# Everyday understanding. As mentioned above, the type of understanding that frameworks should support, and encourage researchers and practitioners to seek and develop, is that of lifeworld of the area. That is, it should be able to (be expandable to) accommodate all the diversity of the everyday life of the area. This can include theoretical, scientific knowledge, but not as its primary form.

Ensuring such characteristics, too, are addressed by philosophy.

1.3.5 A Single Unitary Framework?

As can be seen, there is immense diversity in the ways we understand IS. In each area frameworks offer different capabilities, and some compete for supremacy over others. But the differences are even greater between areas. For example Actor-Network Theory concerns itself with the interaction between people and IS and is usually completely indifferent to whether the IS is programmed using logic programming or object-orientation. And yet there is a link, in that logic programming tends to assume all tasks for which we use computers can be expressed as propositions and predicates in logic, which assumption can constrain our tasks.

There is a move among some [e.g. Shaw and Garlan, 1996] to define ’the’ discipline of information systems as a single, unitary area of research and practice, and therefore to seek a single, over-arching framework by which to understand it. But should we seek a single over-arching framework for understanding the ’whole story’? Jones [2000] calls this ’mission impossible’. Lyytinen [2003] argues that it is neither possible nor desirable: such a quest is ’hopeless’; there is no agreement about what constitutes good theory nor about the appropriate criteria for progress in a field. He points to the incommensurability between positivist and ’postpositivist’ approaches.

This book does not presuppose that we can find one unitary framework for understanding ICT/IS. Rather, it seeks a way of formulating frameworks for understanding each area that cohere with each other by respecting what is deemed meaningful by others or at least take account of issues important in other areas. It is in this way, not by seeking a single framework, that it approaches ’the whole story that is IS’.

But immediately the challenge arises -- and Lyytinen does not seem to have met it -- of how to account for both the incommensurabilities between extant frameworks and the links between areas, and to find a way of converting incommensurability into mutual respect. Such things, yet again, are addressed by philosophy.

1.4 PHILOSOPHY

"One critical, but largely unrecognized aspect in these debates," remarked Lyytinen [2003], speaking of those taking place in the various areas of research, "is the necessity to draw upon philosophical studies as they relate to the nature of scientific knowledge and its foundations. Therefore it is interesting to explore the content of the underlying philosophical argument in these debates and what role they assume to the philosophy as a field of inquiry ." It is more than ’interesting’; it is useful, and even necessary.

Midgley [2000], in Systemic Intervention, gives several reasons why philosophy is necessary, even -- and especially -- when considering practice. Among the reasons he offers are:

# Philosophical assumptions can be used to justify (and critique) practice.
# Philosophy can be used to help define alternatives, and explain why they should be considered.
# Philosophical arguments can assist debates about methodology.
# Philosophy helps us select and defend guidelines for practice.
# Fluency in philosophical debate could help critique and justify intuitive notions and ethical stances in scientific methodology.
# Philosophy helps us see practice in a different light.
# Philosophy reveals why different approaches are incompatible and cannot simply be ’compared’ (he gave the example of utilitarian and rights-based approaches to managing the National Parks in the USA).

While Midgley specially wished to counter anti-philosophical stances in relation to practice, his reasons why philosophy is important can be broadened if we replace the word ’practice’ with ’research and practice in IS’ in most of the above.

1.4.1 Philosophy: a Sketch

Philosophy has a long pedigree, throughout which there have been many ideas on what philosophy is and what its task and role are. To the pre-Socratic Greeks, philosophy linked religious reverence with thinking about the principles of the natural world. Then it became moral and political theorizing, and under Plato and Aristotle, became vast metaphysical constructions. In the mediaeval period, Western philosophy concerned itself with the relation between faith and reason, and was the ’handmaiden of theology’. After the Renaissance philosophy again took up the task of exploring the foundations of physical science, and then turned to examination of the human mind. Kant believed that philosophy’s task was to determine what reason can and cannot do. Hegel, on the other hand, believed that the philosopher’s vocation was to approach the absolute through consciousness. Comte restricted his ’positive’ philosophy to employ only accepted scientific methods, and logical postivism focuses on the activity rather than the content of philosophy. Bergson, in contrast, believed that the deepest and most important knowledge came from what he called philosophical intuition. For Whitehead, the duty of philosophy was primarily to "frame a coherent, logical, necessary system of general ideas in terms of which every element of our experience can be interpreted." [Levi, 1975,p.272]. To the early Wittgenstein "The object of philosophy is the logical clarification of thoughts." Marxist views of philosophy centre on practical issues and stress how class interests affect our world views. To Dewey, philosophy embraced ethics and methodology as well, and he saw its role as helping us understand the unity and interrelatedness of all that is.

It is clear that philosophy is many-sided and can serve many purposes. "Part of what makes it difficult to find a consensus among philosophers about the definition of their discipline" suggests Levi [1975,p.248] "is precisely that they have frequently come to it from different fields, with different interests and concerns, and that they therefore have different areas of experience upon which they find it especially necessary or meaningful to reflect." He gives several examples, drawn across history of philosophy:

> "Thomas Aquinas (a Dominican friar of the 13th century), George Berkeley (a bishop of the Irish Church in the 18th century), and Søren Kierkegaard (a Danish divinity student in the 19th century) all saw philosophy as a means to assert the truths of religion and to dispel the Materialistic or Rationalistic errors that, in their opinion, had led to its decline. Pythagoras in ancient south Italy, Rene Descartes in the late Renaissance, and Bertrand Russell in the 20th century have been primarily mathematicians whose views of the universe and of human knowledge have been vastly influenced by the concept of number and by the method of deductive thinking. Some philosophers, such as Plato or the British philosophers Thomas Hobbes and John Stuart Mill, have been obsessed by problems of political arrangement and social living, so that whatever else they have done in philosophy has been stimulated by a desire to understand and, ultimately, to change the social and political behaviour of men. And still others -- such as the Milesians (the first philosophers of Greece), Francis Bacon, an Elizabethan philosopher, and, in the 20th century, Alfred North Whitehead, a process metaphysician -- have begun with an interest in the physical composition of the natural world, so that their philosophies resemble more closely the generalizations of physical science than those of religion or sociology." [Levi, 1975,p.248]

The implication of this diverse picture is that our response to any statement of what philosophy is must take account of the characteristics of the human being who philosophizes.

Following his quick survey of the diversity of views, Levi [1975,p.248] said, "But if any single opposition is taken as central throughout the history of Western philosophy at every level and in every field, it is probably that between the critical and the speculative impulses." The speculative impulse is often driven by a desire to construct positive philosophical proposals for understanding the nature of things. It builds a proposal, which could then be subjected to the critical impulse. The critical impulse seeks to question assumptions in views held about issues (such as about the nature of computers or what is necessary in IS development). (We may criticise Levi's own speculative use of the word 'speculative' (rather than, for example, 'constructive') as perhaps betraying a presupposition that there are no firm grounds for such construction, a presupposition that will be questioned later.)

Philosophy has a number of main branches, and these are what is necessary to address the issues that constitute FFUs mentioned above:

# Ontology: What is the nature of the world with which those who work in the area engage?
# Epistemology: How do we come to understand and know our area? What constitutes good research? How do we form conceptual structures or a good theory?
# Philosophical ethics: What is Good and Evil in the area: problematic or to be aspired to? What is the root motivation?
# Methodology: What methods should we use in research? In practice how do we overcome problems or achieve our

aspirations?

# Anthropology: What is the role of human beings in the area?

# Critical philosophy: What presuppositions lie at the root of any FFU, and what transcendental conditions are necessary for them to be possible?

Therefore, to consider, discuss, formulate a FFU for any area requires some reference to philosophy. There are many different philosophies, and many views on what philosophy is, so we must clarify what is meant by philosophy. (Even this taxonomy of branches of philosophy might not meet with universal agreement.)

1.4.2 Roles of Philosophy in Information Systems

Philosophy has been applied within the field of information systems for 50 years. At first, artificial intelligence, from the 1950s, referred to philosophy to address the question of whether computers can think, by asking philosophical questions about what thinking and computers are. But the nature of computers is only one among many areas, and the questions raised in the others go far beyond this very limited AI question. The application of philosophy in some areas has until recently been much less frequent and has yet to reach maturity. Philosophical Aspects of Information Systems [Winder, Probert and Beeson, 1997] is one work that contains a number of useful papers in which thinkers try to apply philosophy to some of these wider questions.

In an article in that work, 'Some philosophical and logical aspects of information systems', Robb [1997] suggests [p.8] "To me, the key philosophical question is 'What argument or evidence would justify me in asserting this conclusion to be true?'" But to create FFUs or find philosophical grounding for them requires much more from philosophy than this. We do not start with this or that 'conclusion' to be tested by argument or evidence because not only have we still to determine what conclusions we wish to so test, but we have not even worked out a strategy by which we could do so. Is it appropriate to even seek testable conclusions?

The Cambridge Encyclopaedia of Philosophy goes further, suggesting it is the role of science to find out what is true, but the role of philosophy to find out what truth is. According to this view, philosophy stands back from the sciences to gain a wider, cross-disciplinary standpoint that enables us to understand all such scientific views of truth in relation to each other. While that nicely articulates the difference between science and philosophy, FFUs are concerned not only with truth but also with how we should live in each area (normativity) and with presuppositions that are not 'true' so much as 'held', by a deep, often tacit, commitment.

Lyytinen [2003] characterizes such views as those two as "a positivist, or Popperian concept of what should be the role of philosophy as a reference discipline" and he criticises the assumption that "Philosophy should provide the ultimate language and a logic for the justification of IS knowledge" which lies behind the two views. He continues,

> "This view, as we all know, has been largely contested in the philosophy in the post modern era. Another, 'postpositivist', view of the role of philosophy in the IS field would be to view it as a critical voice in a conversation about information systems that seeks to undermine any quest for a ultimate foundation of the IS discipline. Philosophy deconstructs any language and logic that has been developed for such purposes."

The 'critical voice' is vital to question assumptions and expose presuppositions in each area of IS.

Hodges' [1995] suggests five specific ways in which philosophy can inform information systems research, which have been discussed recently:

# as a reference discipline per se in the sense that it would contribute a subject and a methodology, such as dialectic method (to critically analyse the assumptions of a theory), ethics or general systems theory;
# by helping to lay conceptual foundations on which to build theory, perhaps alongside scientific work;
# epistemologically informed studies of issues in research and practice, since epistemology is a branch of philosophy;
# employing the approaches and insights of hermeneutics and phenomenology in order to develop an understanding of information technology in a broader context; and
# employing the approaches and insights of existentialism to move away from a presumed intellectualism and give due regard to experience and being.

The first three are indeed partly how philosophy is used here, but these still do not fully meet our needs. First, we are not primarily concerned to use philosophy as a reference discipline in itself, since what we are seeking here is not so much a philosophy of information systems, but a philosophy for information systems. We are seeking to find philosophy that can be useful in helping us understand information systems, but we are not restricted to a purely philosophical type of understanding.

Secondly, the foundations we want are not merely conceptual, but must also be normative and presuppositional.

Hodges' third role, Wernick, Shearer and Loombes [2001] suggest, should be broadened beyond epistemology, especially when considering software, its development, evolution and use. Ontology is important, especially to Poli [2001], in order to philosophically inform our understanding of types of technology.

Wernick et. al. have also criticised the last two, suggesting that they are better categorised as examples of the application of specific philosophical directions, while the first three map out areas of information systems research and practice which can be supported by reference to philosophy. But Courtney and Porra [2000] justify the last two as yielding unconventional and imaginative ideas, in contrast to the first three, which they see as reflecting a "somewhat conventional, cognitive processing view of IS research". Moreover, it is phenomenology, and thought derived from it, that drew attention to the lifeworld. The philosophy used in this work is neither phenomenological nor existential, though it has been influenced by both. Rather, it critically questions the very deepest roots of Western thought and it is this critique that generates the "different way of looking at ICT and IS" with which this work opened.

It is these roles of philosophy that qualify it to address the characteristics of frameworks outlined above. As a means of laying conceptual foundations it enables the formulation of frameworks in the first place. As a guide to reasoning it ensures their quality. Transcending different sciences, philosophy enables the whole story to be addressed. Its critical voice makes frameworks open and provides normative direction. Philosophy and existentialism, as seen earlier, made sure the lifeworld is no longer taken for granted. But the desire for a coherence that is distinct from logical consistency, might pose a problem, especially in the light of Lyytinent's desire to "undermine any quest for a ultimate foundation of the IS discipline."

But, just the discussion of IS/ICT should not be constrained by Robb's positivist view, neither should it be constrained a priori by Lyytinen's 'postpositivist' anti-foundationism. As illustrated in the vignettes in the Preface, foundations of an appropriate kind are vital for an everyday, lifeworld approach. Deconstruction will indeed be employed, but Lyytinen's language of "seeks to undermine any quest" speaks of a pre-theoretical dogma against 'ultimate foundations', and the need to adhere to this dogma must be questioned. Lyytinen's dogma seems to have two roots. One is a fear of unimaginative conceptual frameworks (often of a positivist-realist kind) that are inappropriately forced on us. The other is dislike of the battles that have raged between paradigms vying for supremacy over each other.

The approach in this work is to employ a single philosophy, but in a way that should not arouse Lyytinen's fears. It is one that does see a difference between coherence and logical consistency. From it, a diversity of conceptual frameworks and paradigms can emerge and, as mentioned at the start, it does not seek supremacy over others. It will still be the kind of critical voice that Lyytinen desires. It will also ensure proper reasoning, help us stand back from the sciences, act as reference discipline, lay conceptual foundations and stimulate unconventional ideas. It will do these with a respect for the everyday lifeworld. Unlike some recent turns in philosophy, it can cover all the main branches of philosophy listed above: ontology, epistemology, philosophical ethics, methodology, philosophical anthroplogy and critical philosophy.

1.4.3 Philosophical Issues in IS, ICT

The way philosophy is important in formulating frameworks for understanding differs in each area. Table 1.4.3 shows a few issues in each area with the philosophical issues they raise.

Table 1.4.3. Philosophical issues raised by IS issues

Area	IS Issue	Issue in Philosophy
Use of Computers	User-computer relationship	Self and world
	Human activity with computer	Subject-object relation
	Impact in use	Meaning and repercussions
	Benefit versus detriment	Good and evil
	Variety of impacts	Diversity and unity
IS development	Teamwork	Social theory
	Lifecycle methods	Process
	Guidelines	Normativity
	Human creativity	Freedom and intention
	Requirements analysis	Possibility, responsibility
	Knowledge elicitation	Knowledge, epistemology
	Conflict	Perspectives
Nature of computers	Computer as experienced	Self-world relation
	Nature of computer, info	Ontology
	The AI question	Anthropology, ontology
	Hardware, bits, symbols	Meaning and reductionism
Techn'gcl resources	KR languages	Philosophical linguistics
	Types and classes	Universals and individuals
	Inappropriateness	Diversity of reality
ICT as ecology	Validity of ICT as human endeavour	Philosophical ethics and destiny
	ICT as our environment	Structural relations
	Gender issues	Diversity of meaning, norms
	Modern dominance of ICT	Perspectives, progress

It should be clear from this why (with Lyytinen) this work does not seek a single, unitary framework for all areas. But it is philosophy that enables the challenges of coherence between the areas to be faced and to turn incommensurability into respect.

1.4.4 What we need in Philosophy

Different philosophies have been most appealed to in different areas of IS. Though this might to some extent be an accident of history, there might be a more substantial reason: in each area different sets of issues are important, especially in its lifeworld. Each might therefore 'require' a different type of philosophy on which understanding of itself can be grounded. Of the areas mentioned earlier:

# The nature of computers requires a philosophy that takes the nature of things seriously. So ontology is important. It is difficult to see how philosophies of the nominalist tradition can provide a basis for understanding the nature of computers (as opposed to our beliefs about the nature of computers). Moreover, it is not helpful when a philosophy presupposes either that computers and humans are basically the same or that they are so radically different that there can be no comparison between them. It is difficult to see how either rationalist or romanticist philosophies can, ultimately, do more than fall back on dogma when confronted with these issues. We need a philosophy that enables us to discuss the ontic status of both humans and computers, and of the relationship between them.

# The shaping of technology requires not only ontology, but an ontology of diversity. Otherwise, ultimately, we have no basis for discussing the variety we encounter when we take a pre-theoretical attitude to information technology, and are driven back to reductionism. But the ontology we need must also be able to speak of the coherence we experience within this diversity. That suggests we need a pluralistic ontology in which the ontic categories are irreducible to each other and yet there are ontic relationships between them. Reviewing the history of philosophy, we can generally observe that when ontology has been informed by Aristotle's substance-concept, we have usually tended towards reductionism, either of a monist or dualist type. Plato's Ideas concept gives ontic diversity but finds it difficult to account for coherence. Historicism and subjectivism likewise allow for diversity but it is difficult to avoid fragmentation.

# ISD requires a philosophy that can give a genuine account of human freedom, interpretation and perspectives (the insight from SSM) and at the same time possesses a transcendent normativity (the insight from CST). It is difficult to see how a cognitivist stance, or one based on language games, can be sufficient here, even though they might be able to provide a theoretical account of such things in their own terms. Moreover, philosophy should be able to speak to the management aspects of ISD, and not just relegate them to the realm of practice.

# Usage of IS involves humans and IT, and so requires philosophy that acknowledges the possibility of a genuine point of contact between technology and human beings. Being mostly of the lifeworld, with the human being in a social context, usage requires a philosophy that affords dignity to everyday life and to what it means to be fully and socially human. So materialist and rationalist philosophies are unlikely to be helpful. To deal with impact of usage, especially unexpected impact, including on non-human stakeholders like animals or the environment, requires a philosophy that can transcend and yet acknowledge the perspectives of human stakeholders. To differentiate benefit from detriment, especially when both occur, demands an intrinsic normativity within the philosophy, and, again, one that transcends, rather than being derived from, the value systems of those involved in usage. Therefore subjectivist philosophies are unlikely to be sufficient.

# To understand the information society, as a technological ecology, requires a philosophy that enables us to analyse the circular dependency between environment and technology. It must see both inscription and societal structures as meaningful and mutually irreducible. Philosophy that presupposes environment (including society) can be reduced to interactions between individuals is unlikely to be useful.

In the author’s experience, some of which is outlined in the Preface, the radical critical-positive philosophy pioneered by the Dutch thinker, Herman Dooyeweerd, had been able to provide these kinds of insights in all areas. This is unusual because most streams of philosophy cannot -- whether Plato, Aristotle, Scholasticism, early rationalism or empiricism, Kantian, Hegelian philosophy, process philosophy, phenomenology, existentialism, linguistic turn or postmodernism.

1.4.5 Dooyeweerd’s Philosophy

Herman Dooyeweerd, whose philosophy is primarily employed in this work, came from the realm of politics and jural science and was a Christian believer. But he assiduously held that philosophy must not be reduced to either politics or theology, nor to social theory, nor to linguistics, nor to logic, nor to mathematics, nor to any other such special scientific arena. In order to avoid such reduction even in his own thinking, he undertook a transcendental critique of theoretical thought in order to establish the necessary ’transcendental’ conditions that make philosophy as such possible. His conclusions included:

- # Everyday life must be given due respect, as a given, and never treated as a theory.
- # All theoretical thought, including philosophy, is rooted in pre-theoretical (’religious’) stances, and so can never be neutral -- including his own thought.
- # Philosophy is an integrative discipline that enables us to think about the relationships between not only the sciences but also the distinct aspects of our experience, and thus the diversity and coherence of the lifeworld.
- # Philosophy is thus open to issues that are found in all disciplines, such as things and processes, laws and norms, method, knowledge and intuition, the place of the human being, diversity, unity and origins -- that is, ontology, normativity, methodology, epistemology, anthropology etc.

It is this view of philosophy that commends itself to those who wish to formulate LOFFUs for the diverse areas of the coherence that is information systems.

Understanding the above, Dooyeweerd explicitly took the ’religious’ stance of Creation-Fall-Redemption (rather than those of Nature-Freedom or Nature-Grace) and worked out the philosophical implications as a positive philosophy. In doing so, Is and Ought, and Thing and Thought were reconnected after centuries of having been driven apart.

1.5 OUR APPROACH

Faced with such a claim, it is not an option to simply disregard it just because it might be difficult to believe. Rather, Dooyeweerd’s philosophy should be critically examined in order to properly explore its potential. A surface examination will not do.

As mentioned at the start, the standard practice in academic writing is to begin by arguing there is a problem that needs solving, then to make a proposal that solves the problem. This may be appropriate for a proposed theory or methodology but it is not appropriate for proposing a framework for understanding because it is the framework itself that, as a way of seeing things, informs us what is problematic. When we state problems at the start of an argument, we have already adopted a framework for understanding the area in which we are working, even it tacitly, and this might be incommensurable with the framework we are proposing and make it seem irrelevant.

This is especially the case when looking at a whole field. To argue conclusively there is a problem that requires a different philosophy presupposes that philosophy. Therefore, the reader is introduced directly to Dooyeweerd’s philosophy in chapters 2 and 3, to the extent that will be needed to explore each area of IS/ICT. Chapter 2 is an overview that outlines some ways in which Dooyeweerd is different, and offers a dozen reasons why his philosophy should be of interest to us. Chapter 3 introduces portions of Dooyeweerd’s philosophy that will be needed to formulate LOFFUs. Chapter 3 may be treated as a resource for this exploration, but it is designed also to give the reader a little deeper understanding of the ’why’ as well as the ’what’ of Dooyeweerd.

Then chapters 4 to 8 explore in some depth how Dooyeweerd may be used in each of five areas of research and practice to formulate frameworks for understanding. These chapters will use a mixture of philosophical, theoretical and practical discourse, drawing on a range of published material. Each chapter will also discuss how Dooyeweerd’s philosophy can engage with some extant frameworks and major issues in each area. Note that research methodology in each area is not discussed, only research content, strategic directions for research, and some practical devices for use in practice.

The final chapter will reflect on the exploration.

The order in which the FFUs will be developed is not that in which the areas have been introduced above, which begins with the technology and progresses ’upwards’ (what Strijbos and Basden [2006] call a bottom up approach), nor will it begin with societal issues (a top down approach), but rather it will begin with the area in which ICT is most immediately experienced in an everyday manner: the usage of technological artefacts or systems. A framework for understanding usage will be formulated in chapter 4. From there, the discussion moves to other areas quite naturally. Use of such things often leads to reflection on the nature of computers and information, so this is the second area, for which a framework is formulated in chapter 5. The IT artefacts or systems must be developed; this is discussed in chapter 6. Since developers require a set of technological resources, tools and raw materials for this, these must be carefully designed; a framework is formulated for this in chapter 7. Finally, as usage becomes widespread and ICT takes on certain shapes, it takes on a societal aspect, as an expression of our world views, and as something we are happy (or not) to live within and in the light of: chapter 8.

FOOTNOTES

[1] At least 50% [Lyytinen and Hirschheim, 1987], 60% [Cotterill and Law, 1993], 50%, [Whyte and Bytheway, 1996], 60% [Butterfield and Pendegraft, 1996], 75% [Gladden, 1982].

References

Avison, D., BjÃrn-Andersen, N., Coakes, E., Davis, G. B., Earl, M. J., Elbanna, A., et al. (2006). Enid Mumford: A tribute. Information Systems Journal, 16, 343-382.

Bergvall-KÃ¥reborn, B. (2001). The role of the qualifying function concept in systems design. Systemic Practice and Action Research, 14(1), 79-93.

Brachman, R. J. (1990). The future of knowledge representation. In American Association for Artificial Intelligence, AAAI-90: Proceedings of the Eighth National Conference on Artificial Intelligence (pp. 1082-1092). Boston: AAAI.

Burrell, B., & Morgan, G. (1979). Sociological paradigms and organisational analysis. London: Heinemann.

Butterfield, J. & Pendegraft, N. (1996). Cultural Analysis In IS Planning & Management. Journal of Systems Management, 47, 14-17.

Checkland, P. (1981). Systems thinking, systems practice. New York: Wiley.

Cotterill, T., & Law, N. (1993). EIS - A practical approach. Proceedings of the 1993 Hewlett-Packard Computer Users’ European Conference HPECU/HPCUA, Harrow, England. HP Computer Users Association.

Courtney, J. F., & Porra, J. (2000, August). Mini-track on the philosophical foundations of information systems. Paper presented at the Americas Conference on Information Systems, AMCIS 2000, Long Beach, CA.

Davis, F. D. (1986) A technology acceptance model for empirically testing new end-user information systems: Theory and results. Unpublished doctoral dissertation. MIT Sloan School of Management, Cambridge, MA.

Ferguson, N., & Schneier, B. (2003). Practical Cryptography. Indianapolis, IN: Wiley.

Giddens, A. (1993). New rules of sociological method. Cambridge, England: Polity Press.

Gladden, G.R. (1982). Stop the life-cycle, I want to get off. ACM SIGSOFT Software Engineering Notes, 7(2), 35-39.

Glass, R. L., Ramesh V., & Vessey, I. (2004, June). An analysis of research in computing disciplines. Communications of the ACM, 47(6), 89-94.

Hodges, W. S. (1995). Five roles that philosophy can play in MIS research. In Proceedings of the 1995 Meeting of the Americas Conference on Information Systems, AMCIS 1995, Pittsburgh, PA. Atlanta, GA: Association for Information Systems.

Honneth, A., Knodler-Bunte, E., & Windmann, A. (1981). The dialectics of rationalization: An interview with JÃ rgen Habermas. Telos, 49, 3-31.

Husserl, E. (1970). The crisis of European sciences and transcendental phenomenology (D. Carr, Trans.). Evanston, IL: Northwestern University Press.

Jackson, M. C. (1991). System methodology for the management sciences. New York: Plenum Press.

Jacob, M., & Ebrahimpur, G. (2001). Experience vs. expertise: The role of implicit understanding of knowledge in determining the nature of knowledge transfer in two companies. Journal of Intellectual Capital, 2(1), 74-88.

Jones, M. (2000). Mission impossible: Pluralism and ’multi-paradigm’ IS research. Information Systems Review, 1(1), 217-232.

Kuhn, T. S. (1970). The structure of scientific revolutions. Chicago: University of Chicago Press.

Landauer, T. K. (1996). The trouble with computers: Usefulness, usability and productivity. Cambridge, MA: MIT Press/Bradford Books.

Latour, B. (1987). Science in action. Cambridge, MA: Harvard University Press.

Levi, A. W. (1975). Philosophy, history of Western. In Encyclopaedia Britannica (Vol. 14, pp. 247-275). Chicago: Encyclopaedia Britannica.

Lyon, D. (1988). The information society: Issues and illusions. Cambridge, England: Polity Press.

Lyytinen, K. J. (2003). Information systems and philosophy: the hopeless search for ultimate foundations. In J. I. DeGross (Ed.), Proceedings of the Americas Conference on Information Systems: AMCIS 2003. Atlanta, GA: Association for Information Systems.

Lyytinen, K. & Hirschheim, R. (1987). Information Systems Failures - A Survey and Classification of the Empirical Literature. Oxford Surveys in Information Technology, 4, 257-309.

Meister, D. (1999). The history of human factors and ergonomics. Mahwah, NJ: Erlbaum.

Midgley, G. (2000). Systemic intervention: philosophy, methodology and practice. New York: Kluwer/Plenum.

Newell, A. (1982). The knowledge level. Artificial Intelligence, 18, 87-127.

Noble, D. F. (1997). The religion of technology: The divinity of man and the spirit of invention. New York: Alfred A. Knopf.

Poli, R. (2001). ALWIS: Ontology for knowledge engineers. Unpublished doctoral thesis, University of Utrecht, Netherlands.

Robb, F. F. (1997). Some philosophical and logical aspects of information systems. In R. L. Winder, S. K. Probert, & I. A. Beeson (Eds.), Philosophical aspects of information systems (pp. 7-21). London: Taylor and Francis.

Schutz, A., & Luckmann, T. (1973). Structures of the life-world, volume I. Evanston, IL: Northwestern University Press.

Schutz, A., & Luckmann, T. (1989). Structures of the Life-World, volume II. Evanston, IL: Northwestern University Press.

Schuurman, E. (1980). Technology and the future: A philosophical challenge. Toronto, Ontario, Canada: Wedge.

Shallis, M. (1984). The silicon idol. Oxford, England: Oxford University Press.

Shaw, M., & Garlan, D. (1996). Software architectures - perspectives on an emerging discipline. Upper Saddle River, NJ: Prentice Hall.

Strijbos, S., & Basden, A. (2006). In search of an integrative vision of technology. In S. Strijbos, & A. Basden (Eds.), In search of an integrative vision for technology: Interdisciplinary studies in Information Systems (pp. 1-16). New York: Springer.

Walsham, G. (2001). Making a world of difference: IT in a global context. Chichester, England: Wiley.

Wand, Y., & Weber, R. (1995). On the deep structure of information systems. Information Systems Journal, 5, 203-224.

Wernick, P., Shearer, D. W., & Loomes, M. J. (2001). Categorising philosophically-based research into software development, evolution, and use. In Proceedings of the Seventh Americas Conference on Information Systems, AMCIS 2001, Boston (pp.1999-2005). Atlanta, GA: Association for Information Systems.

Whyte, G., & Bytheway, A. (1996). Factors affecting information systems’ success. International Journal of Service Industry Management, 7(1), 74-93.

Winder, R. L., Probert, S. K. & Beeson, I. A. (Eds.). (1997). Philosophical aspects of information systems. London: Taylor and Francis.

Winograd, T., & Flores, F. (1986). Understanding computers and cognition: A new foundation for design. Reading, MA: Addison-Wesley