# Chapter 5.

# A FRAMEWORK FOR UNDERSTANDING COMPUTERS AND INFORMATION

"What is a computer?" and "What is information?" are questions that the reflective user will sometimes ponder. In our everyday experience of them, in all five areas of research and practice, there is something about them that remains the same, whatever application. The discourses in which this issue is central include artificial intelligence (AI), Cyberspace, information theory and some of systems theory. While these offer detailed scrutiny of the issues, it is not to be assumed a priori that the rest of us have an inferior or 'untrue' view.

This chapter explores how Dooyeweerd's philosophy might provide a basis for understanding the nature of computers, information and programs, an understanding that does not depend on their application. It does not attempt to arrive at a single 'best' definition, but rather to provide a foundation for fruitful discourse.

We are not talking about what the computer means to me, subjectively (or us, intersubjectively) nor as enabling particular applications, whether controlling or emancipatory in nature. These issues have been addressed in chapter 4. What is at issue is the nature of the computer regardless of application and of (inter-)subjective meaning. What is it that differentiates computer from, for example, mechanical machine or electronic gadget on one hand, or from other information technology such as writing, printing, film or video on the other? Pre-theoretical experience continues to assert that there is a difference, but it does not allow us to narrow it down to one of the theorized views above.

We seek a framework that enables us to understand computers and information primarily as they present themselves to us in the lifeworld regardless of application. We do not adopt a naïve realism. But neither can a theoretical or technical view be accepted uncritically. There are many suggestions about what a computer 'actually is', for example Schuurman [1980,p.21]: "the basic structure of the computer is this, that a signal in the computer can find its way along either of two alternative, mutually exclusive routes. ... One bit thus affords two mutually exclusive possibiilities -- yes or no, open or closed". Other similarly theorized views include: computer as electronic hardware, computer as symbol-manipulator, computer as agent of equal status with human beings, the only difference being that one is made from silicon and the other from carbon.

But our everyday experience tells us that none of these on its own is sufficient. So is it just a matter of accepting all such views? A difficulty with that is: how do we relate the views or prioritize them in the case of disagreement? Also, how do we relate the nature of computers to that of information or programs?

## 5.1 WHAT IS MEANT BY THE NATURE OF COMPUTERS?

The question, "What is a computer?" is often answered by reference to a definition. Webster’s Dictionary [1975] defines computer as "a calculator esp. designed for the solution of complex mathematical problems; specif: an automatic electronic machine for performing simple and complex calculations". But, for three reasons, such definitions do not help much. First, definitions tend to theorize the type of thing they define and privilege specialist views over everyday experience. This definition is far too narrow to understand the nature of computers today. Second, definitions change. In the 1993 edition of Webster’s Dictionary the second part has been replaced with "a programmable electronic device that can store, retrieve, and process data". Third, definitions presuppose the meaning of such words as ’programmable’, ’electronic’, ’device’, ’store’, ’retrieve’, ’process’ and ’data’, which all have to be understood if the definition is to help us understand the nature of computers. Because of these problems, the nature of computers becomes a matter for philosophical analysis rather than definition.

### 5.1.1 Philosophical Understanding of the Nature of Things

The question of the nature of computers, information, program, etc. is part of the wider question of the nature of Being or Existence: what does it mean to be a particular type of thing (a computer). On what basis may we differentiate one type of existence from another?

It may be no coincidence that those in this area often refer back to Greek thought, or early modern thought which was influenced by Greek thought, because it was the Greeks who took seriously the ontological questions about the nature of things. Under NGGM only the sacred was worthy of our philosophical attention. Under the Nature pole of NFGM, physics and mathematics seemed to give the complete answer to the nature of things, so no further exploration was required (the Romantics were dissenting voice). But this obliterated human freedom, so a swing to the Freedom pole occurred. Under the Freedom pole, post-Kant, it is presupposed that ’things in themselves’ cannot be known, so ontology was replaced by, or made subservient to, epistemology.

As a result, it is fashionable in some areas of IS, especially those of ISD or societal matters, to eschew ’essentialism’, and thus seemingly to forbid any attempt to discuss the nature of computers as such. Extreme versions of this view would, at a stroke, wipe out this whole area of research and practice as completely misguided. But that approach fails to do justice to our everyday experience, that there is such a type of thing as ’computer’ and ’information’.

Less extreme versions might allow discussion of the nature of computers, but would always derive this from our (inter-)subjective views of what a computer is, which are diverse and open-ended. But such views do not allow the thinker to take seriously that the computer presents itself to us as something with an integrity beyond our (inter-)subjective beliefs about it, or our application of it. Moreover, they provide no grounds for discussing the real potential and limitations of what is computer, such as expressed by the artificial intelligence question of whether computers can truly understand, nor for either hope or despair regarding those.

But this chapter is about understanding computers and information regardless of their subjective meaning to us or their application, and thus what might be possible in future, especially with regard to . Therefore, the discussion in this chapter will be ontological rather than epistemological in character.

So those who wish (still) to explore the nature of computers revert to FMGM thinking because it seems to be the only one available so far. It emerges in two main ways in IS. Questions about the nature of computers are couched in terms of Brain versus Mind (though other versions will be discussed later). Systems theory's notion of emergence, such as in Wilber [2000], may be seen as a multi-level version of FMGM.

But Dooyeweerd, in his complete break with immanence philosophy, which underlies FMGM, NGGM and NFGM, has a different way of understanding Being, and a different approach to ontology that does not need to be subsumed into epistemology, yet answers the challenges that fired Kant and all since him (see §2.4.5).

### 5.1.2 Some Issues

Nevertheless, the nature of computers is closely bound up with human experience. It has links with usage, discussed in chapter 4, but mainly with HCI rather than either ERC or HLC. This provides a useful link between these two areas of research and practice: how the user experience computers in everyday living should at least be commensurate with how we understand the nature of computers, information and program as such, even though not identical with it. Therefore, at least, HCI may be a good place to begin in order to discern a number of issues.

The first issue is diversity of our experience of computers. The computer presents itself to us in many ways -- as hardware, as pixels and sound, as symbols, as useful content, and so on. A certain science might focus on one of these, but it cannot account for them all. Newell [1982] recognised this, and suggested that computer systems are multi-levelled beings, understandable in a number of distinct ways. A similar approach may be seen in discussions of the relationship between data, information and knowledge.

The second issue is potential of computers; on what basis may we explore and discuss the potential of computers? What will they be able to do, and not do, once current limitations are overcome? What is the status of a statement like "Computers can think"? Will we end up with Cyberspace as a kind of reality that is completely independent of matter and flesh? Or are there fundamental barriers to either of these? If so, or if not, why?

The third issue is the innards of the computer. Our everyday understanding of the computer tells us there are files ’in’ the computer. There are bits and bytes, numbers, data structures, etc. This chapter on the nature of computers is (hopefully) ’in’ my computer. But how can I say they are really ’in’ the computer, if, when I open the case, all I see is the motherboard, cables, fan, etc.? There are three practical reasons why it is valid to believe these are ’in’ the computer:

- One is the rather obvious reasoning that since it is we who placed the files on disk, and can retrieve them, so they must be there, and since it is we who input various data, so that data, or some result of processing it, must be there.
- The second is by analogy with human functioning. For example, human semantic memory is often likened to a data structure inside the computer.
- The third is that since we see something on the screen (in each aspect) then it is reasonable to believe that something ’inside’ the computer ’caused’ this to be displayed. For example, I see shapes formed of pixels in various colours, suggesting there is a bitmap somewhere ’in’ the computer, which was ’caused’ by various bit manipulations, or I see the number 3984, suggesting there is a numeric variable in the program with that as its value, or, knowing that 3984 is the height of Ben Lawers (a mountain in Scotland), I can say that the height of Ben Lawers is held inside the computer.

But these do not address the nature of innards, and they are not fundamental and may still be argued against philosophically. So can we find any philosophically sounder understanding of the innards? I have not encountered so far any attempt to do this using conventional streams of philosophy, possibly because approaches based either on sensory experience make innards irrelevant while those based on a substance-idea (see below) presuppose innards and thus cannot make it into a philosophical question to be asked. It is merely assumed to be the case that such things are ’in’ the computer (though there is debate about what these are), and the very posing of the question is taken as evidence of the inferiority of the pre-theoretical approach. So the question still remains open: on what basis can we validly say that such things as bits, numbers or content are ’in’ the computer?

The fourth issue is persistence and change. Though continual replacement of old technology has been the norm over the past 30 years, and legacy systems is usually a term of disparagement, there is no fundamental reason why this should be so. Increasingly, people are coping with the technology they have, merely upgrading it, so there is a need to understand persistence and change. ’My computer’, an Amiga 1200 on which this book was written, began life in 1993, but since then it has undergone change: added memory, added SCSI, added faster processor, replaced hard disks, removed SCSI, new motherboard, new keyboard, new mice, new version of the operating system, added software -- but it is still ’my computer’. How may we account for this experience of identity despite change?

Perhaps more important is software and data. This book, as a computer file, has been rewritten five times since it began as a project in 2003, has completely new structure, argument and even purpose, might spawn another book, and probably there is none of the original text remaining -- yet it is still the same book. Such questions are important not only with distinct projects but also in legacy systems. Acknowledging the continuing identity of a thing is important, not just to settle arguments, but for legal reasons and to instil vision that inspires.

The fifth issue is norms for computers: what is a 'good', 'true' or 'real' computer? How can we make it more 'true'? In everyday life, such questions are frequently asked of things.

These issues all concern the structure of the thingness of the computer (and information and program): what is necessary for a thing to be a computer?

### 5.1.3 The Need for Dooyeweerd's Approach

"As far as I know," concluded Dooyeweerd [1984,III,p.53], "immanence philosophy, including phenomenology, has never analysed the structure of a thing as given in naïve experience." It is not that nobody has tried to analyse the structure of things, but that they have not done so as they present themselves to us in everyday experience. In an extensive survey [1984,III,p.3-53], Dooyeweerd examined a number of answers that have been offered to the fundamental question of what it means for a thing to be that type of thing.

- # Fiction: that there is no 'thingness' as such, only a fictitious union of sensory impressions; that view is obviously unsatisfactory.

- # Reduction to sensory function [p.28-36,102]: that we can understand the nature of things by, and only by, sensory experience of them; that prevents any understanding of innards, and of norms.

- # Reduction to mathematical-logical relations, as in Russell [p.25]; this provides no basis for understanding the diversity things present to us, without importing other meaning a priori; it cannot even differentiate between hardware and software.

- # Aristotelian substance concept [p.17-18]: there is a 'computer-substance' of which all computers are made (for example silicon electronics plus programming, or a certain type of causality [Searle, 1990]); this makes it difficult to account for the variety of our experience and consideration of potential becomes open-ended speculation that degenerates into dogmatic positions (such as on whether computers will ever be able to understand; see later). This view also contains an antinomy [p.11,17].

# Process philosophy, a variant of the above which emphasises the dynamic over the static in substance; Dooyeweerd saw it as a "meaningless alternative" [p.18].

# The "modern Humanistic concept of substance, following Descartes", which Dooyeweerd dismissed with [p.26] "Whereas the Aristotelian idea ... was at least intended to account for the structures of individuality as they are realized in the concrete things of human experience, the modern concept of substance was meant to eliminate them."

# Kant's identification of "the 'things' of naïve experience ... with the Gegenstände of natural scientific thought" [p.28]; Dooyeweerd continued, "This procedure immediately resulted in the elimination of the datum of naïve experience." As a result, we can never know the 'thing in itself' even in principle, and so ontology is absorbed into epistemology.

# Heidegger's account of being-in-the-world; though perhaps an attempt to provide an everyday account, cannot account for innards nor for normativity.

Dooyeweerd, contrary to this, believed we can discuss the nature of things as they present themselves to us in everyday experience, but in a different way, based on the CFR ground-motive.

## 5.2 A DOOYEWEERDIAN APPROACH TO THE NATURE OF COMPUTERS

As explained in chapter 3, Dooyeweerd believed our starting point for understanding the nature of anything must be that everything functions and exists within a cosmic framework of Meaning and Law-Promise (see §2.4.4). So we do not ask, first, "What is computer?" but "What means computer?", not "How does computer behave?" but "What law enables computer?" Therefore, instead of seeking to identify a self-dependent essence or substance (nor even process or causality) that is 'computerness' or 'information', by reference to which all discussion within this area can occur, such things as being and behaviour are derived from Meaning/Law, of which we may delineate a number of distinct spheres (viz. the aspects), and what differentiates the thing that is computer from other things (such as lump of silicon, electronic device, or digital device) is the internal structural principle, which involves all aspects led by the qualifying aspect (see §3.2.5).

To illustrate the general approach that will be adopted, consider the book you are holding.

# It is a lingual thing: a discussion of philosophical frameworks for understanding information systems.
# It is a formative thing: a structure of chapters, paragraphs, etc.
# It is a physical thing: half a kilogram of paper.

- # It is a juridical thing: someone's property.
- # It is an economic thing: a product with a cost.
- # It is a pistic thing: the author's vision for how we should understand information systems.

The being of a thing is an interlacement of several 'aspectual beings', each of which is a reification of its meaning in that aspect. The nature of computers can be understood in a similar way, but a closer analysis of this is needed.

### 5.2.1 In Relation to Human Beings

"In veritable naïve experience," Dooyeweerd believed [Dooyeweerd, 1984,III,p.54], "things are not experienced as completely separate entities." To understand computerness, therefore, we must understand it in relationship.

In chapter 4 three ways in which the computer can relate to its user were distinguished: as something with which the user interacts (HCI), as represented content (ERC) and as an artefact used as part of human living (HLC). To understand the nature of computers regardless of application, only HCI provides a useful starting point because with both ERC and HLC the main (qualifying) aspect varies with the application. HCI is qualified by the lingual aspect. Perhaps this this is what Winograd and Flores meant when they said [1986,p.78] "Computers do not exist, in the sense of things possessing objective features and functions, outside of language."

Chapter 4 gave two main facts about the structure of HCI as understood from a Dooyeweerdian viewpoint:

- # It is multi-aspectual human functioning concerned with the human being's direct experience of the computer (specifically via its user interface (UI)). This multi-aspectual functioning is led by the lingual aspect.
- # In all post-physical aspects the computer functions as object as part of human functioning, while in the physical aspect it may function as subject.

That the latest subject-functioning aspect of the computer is the physical is what makes computers so useful. The later aspects, especially from the analytic, are non-determinative and so all subject-functioning in them will be non-determined. But, barring quantum effects, the laws of the physical aspect are determinative, and as a result the behaviour of the computer is determined and predictable. That means that, in computer technology, humanity has at its disposal the possibility of meaningful functioning in the later aspects that is reliable and predictable. Other implications of the difference between subject- and object-functioning will be considered later.

Here the cosmic meaning which the computer has in all aspects, whether as subject or object -- which will be called meaningful-functioning -- can reveal the nature of computers.

### 5.2.2 Human Experience of the Computer as a Whole

Table 5.2.2 shows some examples of how the computer functions in each aspect in relation to the human being, regardless of its application. These are ways in which we encounter the computer in our everyday life as users and/or developers.

Table 5.2.2. Aspects of Computer

| Aspect | Functionings of Computer Meaningful in Each Aspect | | |
|---|---|---|---|
| Quantit'ive | A lot of stuff on screen | | One computer |
| Spatial | Screen layout, size | | Space taken up on desk |
| Kinematic | Animation on screen | | Fan air flow |
| Physical | Light emitted from screen,<br>Vibrations from speakers<br>Pressure I exert on mouse | Electromagnetic<br>fields | Force exerted on desk |
| Biotic / Organic | Activates nerves in ear, eye<br>My hand pushes mouse | Voltages | Repetitive strain<br>injury |
| Psychic / Sensitive | Colours, shapes on screen,<br>Sounds from speakers<br>Key hits, Mouse moves | Memory bits,<br>signals | Case is beige colour,<br>Fan noise |
| Analytic | Icons, numbers<br>Mouse gestures | Pieces of data | |
| Formative | Tables, lists, paragraphs<br>Syntax of my command | Data structures<br>algorithms | How the screen, wires,<br>keyboard are<br>connected, arranged |
| Lingual | Represented Content | | Manufacturer's, logo<br>Labels on sockets |
| Social | Cultural implications | | Sound from computer<br>annoys others in office |
| Economic | Limited screen area<br>Max keyboard rate | Limited memory size | Cost of purchase |
| Aesthetic | Style of UI | | Style of case<br>as decor in room |
| Juridical | Appropriate expression of info | | Ownership of computer |
| Ethical | ? | | |
| Pistic | | | |

The left-hand column refers to what we experience directly via the UI, the middle column refers to what we have learned to believe is 'inside' the computer, and the right-hand column is sundry other functionings, which have little to do with it as a working piece of information technology and more as any other material artefact that is owned and is part of our lives. Differentiating in such a way accords with our intuition, but on what philosophical grounds may we do this?

Our experience of or in the left-hand revolves around the qualifying aspect of HCI, which is usually the lingual. For example, its screen content is understood (lingual functioning), the text and graphics on the screen are structured (formative), they are taken note of as types of data (analytic), the text characters are seen and recognised by virtue of being known fonts (psychic), and the screen emits light (physical). In the anticipatory direction, the understanding is facilitated by cultural connotations (social), there is a limit to the amount of text on screen (economic), and so on.

But the right-hand column shows the computer functioning in ways that are not always necessary to serve the leading aspect. For example, being visible to us (e.g. from the back) is psychic functioning, but not relevant to its lingual functioning above. (The computer might serve a contingent function, such as making money for its manufacturer or acting as paper weight. The success of Apple computers might in part lie in the harmony between the two aesthetic functions, which itself is meta-aesthetic.)

### 5.2.3 The Innards

The middle column refers to things 'inside' the computer, and requires more elaborate treatment.

Dooyeweerd offers us a basis for considering innards. He addressed the question of things 'hidden' from our naïve experience. First, he argued that even though it is through sensory functioning that most of our experience comes, "Naïve experience ... is by no means restricted to the sensory aspect of its experiential world." [Dooyeweerd, 1984,III,p.102, footnote] So the fact that we cannot directly see or hear bits, numbers, content, etc. in a computer does not rule it out from being part of our naïve experience. It does not mean such things are in any way a mere theoretical abstraction from the sensory or physical.

But, if they are hidden how can we experience them? Our sensory function can be 'opened' by means of techniques and technological apparatus. Dooyeweerd gave examples of microscopes, telescopes and using developed physical theory (for experiencing cells, galaxies and atoms, respectively; there are various reasons why things may be 'hidden'). Our experience of such things may be indirect but it is still everyday and not theoretical; even though such techniques and technological apparatus are the product of the theoretical attitude, their concrete actualisation in life brings them into the sphere of our naïve experience. Specifically, we have apparatus by which we may experience the innards of the computer indirectly in each aspect. For example:

# Organic aspect of hardware and electronics: oscilloscope, etc. and a circuit diagram.
# Psychic aspect: memory dump software
# To interpret this memory dump at the analytic aspect, a lot of memory-dump software also shows the bytes interpreted as ISO characters, or via some other bit-to-data coding.
# To interpret structure and purpose of this data (formative aspect) the user needs to know the program structures, or a tool that knows this (for example the amazingly useful program called Structure Browser, which allows its user to move around the structures of the Amiga operating system).
# In the lingual aspect, the program has meaning only insofar as the user knows the parameter interface and what the program is supposed to do in terms of its application.

Thus we have a soundly philosophical way to underpin our intuitive belief that the computer 'contains' various things inside it, and these are meaningful at various different aspects.

### 5.2.4 Excursus: Reinterpreting the Biotic-Organic Aspect

To Dooyeweerd, the kernel meaning of the biotic-organic aspect is life functions, vitality. But we have reinterpreted this somewhat from the customary life functions, to that of hardware components. What justification do we have for this? (This section is mainly of interest to Dooyeweerdian scholarship in that it shows how the Dooyeweerdian view might be taken further, and may be skipped.)

We could argue that our framework for understanding the nature of computers needs to differentiate hardware from both (physical) materials and (psychic) bits and signals, and that between these two aspects of these two lies, very conveniently, an empty slot which is the biotic aspect. But philosophical convenience is no good reason.

It is obvious that the computer, not being alive, does does not function as subject in the biotic aspect. But it is also difficult to see how the computer functions as object in this aspect, since it is not a means of life for any living thing, nor is the content of the program it is running necessarily about a biotic topic. We seek to identify the biotic meaning that is germane to the computer being a computer, if such exists. Dooyeweerd posed similar questions in his discussion of Praxiteles' sculpture of Hermes and Dionysus [Dooyeweerd, 1984,III,p.112ff.].

We have three reasons for treating the hardware of the computer as its biotic aspect. The line of reasoning goes as follows (remembering that the computer functions as object rather than subject in the biotic aspect).

- # We start by considering the biotic aspect of our interaction with the computer (HCI).

- # At the biotic level of the human being, we speak about organs like stomach, fingers, eyes, ears, in contrast to the chemical-physical material of which these on one hand and feelings, sensations and motor impulses (psychic-sensory aspect) on the other.

- # With what, of the computer, do organs engage? Hands grasp a mouse, rather than moveable plastic, and eyes see the screen, rather than light. Our organs engage, not with physical material, but with manufactured components.

- # Therefore perhaps it is valid and useful to say that the discrete, manufactured hardware components of the computer function as biotic objects to the user's biotic subject-functioning.

A secondary reason to support this view, which might arouse controversy in Dooyeweerdian circles, is that one important feature of living things is that they maintain a distinctness from their environment, an active equilibrium state different from the environment, even as they interchange physical material and energy with it -- unlike many physically qualified things like areas of rock or river currents. And organisms repair themselves and thus maintain their integrity as organisms. Computers too maintain a distinct active equilibrium and maintain their integrity (e.g. by checksums built into memory cells).

Naïve experience -- to which we must always listen sensitively in a Dooyeweerdian approach -- offers a third reason: for several decades it has seemed meaningful to compare and contrast machine with human body, which suggests they lie within the same sphere of meaning (i.e. aspect).

For these reasons, therefore, the hardware that is the computer will be seen here as its biotic object-aspect, but to differentiate between biotic aspect involving living things, and this hardware aspect, we will use the word 'organic'. However, this is contentious and requires further debate within the Dooyeweerdian community.

### 5.2.5 Aspectual Beings that Constitute the Computer

Sometimes it is convenient to talk solely in terms of what is meaningful about the computer within each aspect, but usually it is more convenient to talk about 'things' related to the computer, i.e. nouns rather than adjectives or verbs. If Being is complex and founded on aspectual Meaning, then the computer exists in many different aspects -- it is many 'aspectual beings'. This accords with naïve experience but is foreign to most theoretical ways of understanding Being. Dooyeweerd held [1984,II:418-419] that "On the immanence standpoint it is impossible to recognise the modal all-sidedness of individuality." One possible exception to this is Newell's multi-level view of computers, discussed below.

Table 5.2.5 shows a host of aspectual beings of the computer: what we deem 'things' or activity that are meaningful about the computer in the physical to lingual aspects. The aggregations from left to right, as well as the vertical relationships across aspectual boundaries are mentioned later.

This illustrates quite clearly Dooyeweerd's claim that aspects are modes of being. The computer as such exists as materials, as hardware components, as memory etc., as raw pieces of data of various types, as structured data, as applications content. It exists as all of these at the same time, not one after the other. Aspectual beings are merely ways in which the computer is meaningful reified into things.

Six aspectual beings have been identified for the computer, and five for the book. In fact, Dooyeweerd contended, all things are meaningful in all aspects, though sometimes only latently. There is nothing which has an aspect missing. If there were, then that thing would never be able to be an object in that aspect. (This seems to be something of a dogma to Dooyeweerd, and some might question it. But it is a reasonable dogma, because it reflects our everyday experience, and we will adhere to it here.) For example the aesthetic aspectual being of the computer is what its lingual aspectual being anticipates and makes possible, and yet also what serves to either serve or undermine that lingual function. This does not refer to any aesthetic use to which the computer is put, which was differentiated in chapter 4 as HLC, but the aesthetic aspect of HCI, which includes the style of the user interface.

Table 5.2.5 Aspectual Beings of Computer

| Aspect | Atomic | Aggregations | | | | |
|---|---|---|---|---|---|---|
| Physical | Fields, Quanta, Waves, Particles | Atoms, Crystals, Energy troughs | Doped Silicon, Conductance, Light-emitting phosphor | Behaviour of P-N junctions Light emissions | Active whole computer | Electric connect to grid |
| Organic (Biotic) | Electronic component<br>Conductor<br>Voltage, current<br>Mechanical component | Circuit board<br>Speakers<br><br>Mech'l assembly | Larger assemblies (disk, mouse) | Computer as hardware | | |
| Psychic | Bit<br><br>Signal<br>Coloured pixel<br><br>Sound<br>Key press, etc. | Byte, word<br>Register<br><br>Shape, pattern, texture<br>Sound FX<br>Gestures: click,, d-click drag | Alloc'd memory machine instr'n<br>CPU<br>Bitstream<br>Shapes, bkgnd, animation<br>Complex sounds | Machine code program<br><br>TCP/IP<br>Window, screen<br><br>Bkgnd music<br>Sequence of gestures | | |
| Analytic | Datum: integer, letter, etc.<br>Data change<br>Visual or aural datum<br>Enter a letter<br>Select | Set of data<br><br>DB transaction<br><br>Data input | All the data and its manipulatio | | | |
| Formative | Relationship between data<br>Transformation<br>Relshp bet things displayed<br>A whole user interaction (qn, ans, help, etc.) | Record, data structure<br><br>Algorithm, inference engine<br>Diagram<br><br>Rational sequence of user interactions | Database<br>Program<br><br>Document, website<br>User session | | | |
| Lingual | Symbol with its signifiation | Piece of knowledge | Knowledge gained or refined<br>Virtual reality | Application, multimedia title game, etc. | | |

It might be asked why the digital bit is not of the analytic aspect of distinction, and has been linked with the psychic aspect. There are two reasons. The bit need not be digital but could be continuous as in analog computers below. And the bit equates, in human functioning, to activation of neurones (psychic) rather than to mental concepts (analytic).

### 5.2.6 Analog Computers

The discussion thus far has assumed digital computers. But this Dooyeweerdian approach allows for analog computers too. In analog computers, continuously variable voltages and currents signify numeric quantities in what seems a more direct way than by digital coding (e.g. 0 -- 5 volts might map to a level of activation. This assigning of lingual meaning (semantics) to voltages involves the intervening aspects, but in a rather simple one-to-one way, with the result that it can be difficult to separate out the different aspects.

- # 'Organic' aspect: a voltage or current (e.g. 4.2 v)
- # Sensory-psychic aspect: the voltage-level as a level of activation (e.g. as proportion of voltage-range 1-5 v)
- # Analytic aspect: the sensed thing as an quantity (e.g. 80% of range)
- # Formative aspect: the distinct variable with a purpose, and among other voltages
- # Lingual aspect: the semantic meaning of the purposeful, related variable, such as what angle to raise the gun.

Though most of the discussion in this work will be in terms of digital computers and information, it should always be borne in mind that most can be extended to analog technology.

### 5.2.7 Meaningful Wholes

These aspectual beings do not have any existence apart from the whole that is the computer. The term 'meaningful whole', or just 'whole', will be used to refer to the entire thing in its unity as a thing, as it appears to us in everyday life. A thing like a computer system is a meaningful whole. It is the meaningful whole that presents itself to us first to our everyday experience. The coherence of such wholes is possible, philosophically, because we presuppose the harmony of the aspects (see §3.1.4).

(An early version of the proposal developed here may be found in Basden and Burke [2004], which addresses the related question, "What is a document?" It finds a similar dynamic, multi-aspectual meaningful whole, and also discusses aspects of responsibility. The reader will find a different slant on these issues therein.)

'Aspectual being' is, however, not a phrase that Dooyeweerd used. He seemed not to refer to the actual being as such, but the structural laws or principles that make such beings possible. He spoke of 'individuality structure', but this usually referred, not to the concrete being but to a general type of such beings, to the law-side 'internal structural principle' (§3.2.5) by which such beings might exist. We want to reify aspects of the thing, so that we can use nouns or noun-concepts when thinking about it, rather than having to restrict ourselves to other parts of speech. Thinking of a thing's aspectual beings is largely equivalent to thinking about its aspects, so the two may usually be used interchangeably. But they have different characteristics and sometimes we will find one more helpful, sometimes the other. Thinking about aspectual beings enables us to consider relationships among things in a computer.

### 5.2.8 Relationships Among Things in a Computer

Aspectual beings cannot be seen as parts of the whole, not in the way pages or chapters are parts of the book. So how do these things relate? Table 5.2.5 listed many aspectual beings of the computer, with two types of relationship between them, horizontal and vertical.

### 5.2.8.1 Relationships between beings within an aspect

Within each of the aspects except the physical we have several lists of things, beginning with basic or atomic things that cannot be subdivided in this aspect, followed by various degrees of aggregation of these things. This is the part-whole relation, which Dooyeweerd characterised as being between things qualified by the same aspect; for example a bit is part of a byte, which is part of a chunk of allocated memory, or an integer is part of a record which is part of a table in a database. This enables us to understand aggregations of things as in Table 5.2.8.

But it cannot elucidate the relationships between things in different aspects. There is something wrong in saying that a bit pattern is part of an integer or text string: a category error.

*5.2.8.2 Relationships between beings of different aspects*

The relationship between aspectual beings and their whole is foundational enkapsis. Dooyeweerd illustrated by reference to Praxiteles' sculpture Hermes and Dionysus: the relation between the sculpture and the block of marble from which it is made. Whereas Praxiteles' sculpture involved two main aspects (physical marble, aesthetic work of art), in a computer, at least six aspects are involved, from physical to lingual -- these are the aspectual beings we identified above: material, hardware ('organic'), bits (psychic), pieces of data, structures and processing, and content.

In foundational enkapsis, foundational inter-aspect dependency (§3.1.4) plays an important part. Each aspectual being depends on those of earlier aspects in order that it may be 'implemented': profit level is 'implemented' in numbers, which are 'implemented' in binary-coded bit patterns, which are 'implemented' in voltages, whose components are 'implemented' in silicon. A similar account, in both directions, may be made of the user interface screen, beginning with phosphor and glass to make a cathode ray tube.

### 5.2.9 Implementation

When we use one aspect to implement the next, we 'add' the meaning of each aspect to what we already have, and it is a different kind of meaning with each move. Thus, starting with the

- Physical aspect of materials (like silicon, phosphor, glass),
- to implement the 'organic' aspect of hardware components like IC chips, collect the materials together into distinct 'organs' or hardware components.
- To implement the psychic aspect, interpret certain voltages etc. as digital states (e.g. 5v = 1, 0v = 0), and the components that hold those voltages as memory cells, registers, etc.
- To implement the analytic aspect, add a coding system like ASCII or binary (e.g. bit pattern 01100001 is the number 97 under binary coding).

- # To implement the formative aspect, add structuring and processing (for example, the number 97 might be subtracted from 114 to yield a new number, 17).
- # To implement the lingual aspect, add semantic meaning (e.g. these three numbers might represent expenditure, income and profit).
- # To implement the social aspect, add the cultural connotations of these numbers (e.g. "Profits only 17M? Hmmm: a bad risk").

Because of the fundamental irreducibility in meaning between aspects (§3.1.4), how an aspectual being in one aspect may be implemented in an earlier aspect is not determined. This provides a philosophical account for several things that are usually taken for granted:

1. It gives implementors much freedom, at every level:

- # Some computer components are made of Gallium Arsenide instead of Silicon.
- # Different voltage levels carry the bits 1, 0.
- # The bit pattern 01100001 is the letter 'a' under the ASCII code.
- # The number 97 might mean, not expenditure, but the number of students in my class.
- # Profits of 17 might indicate a healthy rather than weak performance.

As a result, the same software can run on hardware from different manufacturers, why it is possible to make advances in hardware without necessarily upsetting the working of the software

2. It enables virtual data (as it is called in database circles). What is one being in one aspect might be many beings and even many activities in the earlier aspect without an actual 'static' thing. For example, the information 'profits last year' might not be stored in the database or computer as a single datum (analytic aspect) but, whenever it is called for, a quick calculation of profit is made on the basis of two other figures, income and expenditure. Such 'virtual data', though a single lingual aspectual being, is stored as multiple analytic aspectual beings together with the formative aspectual functioning that is the subtraction process. Another example: whereas in most computers a memory bit is implemented as a static electric charge, in one digital system the author once worked on in the mining industry, where there is much electric interference, the single bit was implemented as a phase change in alternating current waveforms.

3. But it makes it impossible to interpret something in one aspect unambiguously in the next aspect if we do not import meaning from that aspect. For example, given the bit pattern 01100001 we cannot tell whether it implements the quantity 97, the letter 'a' or anything else, unless we already take into account its meaning within the analytic aspect. Likewise, we can have no idea what a piece of program code (analytic and formative aspects) without meaningful names or comments stands for (lingual aspect) -- the curse of programs written without comments!

4. It makes randomizing and file compression possible. Randomization involves an operation on the bit pattern of a number that makes sense in the psychic aspect (for example involving exclusive-or) but makes no sense at the analytic aspect. File compression involves bit-level (psychic aspect) operations that alter the coding without altering the analytic data: for example compressing a file encoded as ASCII characters as a ZIP file.

5. It also enables us to understand errors of various types. What is an error at one level is explicable at the next lower level. For example, if a program's memory cell is overwritten by another program (such as a virus) then, from the point of view of the psychic aspect, all that has occurred is that a bit pattern has changed, and in principle we could know which program did this. But at the analytic aspect of data, the value in the variable has suddenly changed, and the change is completely inexplicable even in principle.

## 5.3 INFORMATION AND PROGRAM

### 5.3.1 A Dooyeweerdian Understanding of Data, Information and Knowledge

Surprisingly, perhaps, there is still debate about the relationship between data, information and knowledge. It has been made a current issue by knowledge management in organisations, where one speaks of the collection of databases or a data warehouse as 'the company's knowledge base'. Knowledge and information are somehow 'in' the data warehouse.

Alavi and Leidner [2001,p.109] suggest "data is raw numbers and facts, information is processed data, and knowledge is authenticated information". Checkland and Holwell [1998] review a number of views to suggest an extra link: capta. Then "the attribution of meaning in context converts capta into something different, for which another word is appropriate: the word 'information'" [p.90]. Information then contributes to "larger-scale, slower-moving knowledge". ('Capta' was actually used by Langefors [1966] to denote something different: what is 'captured' from perceptions, and which then becomes information.)

Checkland and Holwell's account presupposes processing and a temporal sequence from data through to knowledge. This might be adequate for the example they give of a manager accessing a database of sales figures, selecting them (capta), applying context (information) to contribute to knowledge of the market. But there are four problems. One is: is there always a temporal separation? That assumes there can be such a thing as 'data' that is not yet information or knowledge. Though the data might not yet contribute to that manager's knowledge without processing, it itself is not 'just data'. It is a sales figure, which was at one time information and knowledge for someone else. Second, they presuppose a distal relationship (analytical) between the user and the data. But here we need to account for the proximal relationship too, the immediate, perception of what is on the computer screen, which is, simultaneously to us, shapes, digits, numbers, information and what it signifies. Third, Tuomi [1999] put forward an "iconoclastic argument" that "the often-assumed hierarchy from data to knowledge is actually inverse: knowledge must exist before information can be formulated and before data can be measured to form information." Fourth, their account starts with data as pre-given, but how does it relate to the the medium, whether bits in computer memory or marks on the page?

A Dooyeweerdian view, similar to that developed for computers above, can overcome these problems. Information, data, knowledge, etc. are the same thing, just seen from different aspects:

- # Pscyhic aspect: 'Bits' and 'states' refer to Shannon's and Bar-Hilell's views of what they call 'information', to differentiate it from its biotic or physical medium. They concern pre-conceptualized bits, with signals and signal paths. On screen this is pixels, on paper this is visible deliberate marks, and heard, this is sound.

- # Analytic aspect: 'Data' may usefully refer to what we have called raw pieces of data.

- # Formative aspect: 'Information' may usefully refer to data as part of something which has been processed or structured.

- # Lingual aspect: 'Knowledge' may usefully refer to what the information is about (though, we use 'knowledge' here in a different way from Dooyeweerd in chapter 3).

- # 'Wisdom' may be added, refer to our taking all aspects into account when we consider what the information is about [De Raadt, 1991].

This links these to the medium, it does not presuppose a temporal sequence or processing, nor a distal relation, but rather can allow for a Gestalt immediacy in which the bit-perception is the data is the information is the knowledge. The sales figures are bits, data, information and knowledge simultaneously. There is nothing in a database or a piece of literature that is 'only' data without being at the same time all the others. This view is thus comfortable with Tuomi's iconoclastic argument, in that our knowledge is involved in this immediate experience. Bits anticipate data, which anticipates (processed and structured) information, which anticipates signification (knowledge).

(The manager's processing of the sales data is not a matter of making it into information that it was not before, but rather of creating something new by analytical and other functioning.)

But the picture is further confused by knowledge management issues that have arisen over the past decade. Walsham [2001], after discussing Checkland and Holwell's view, moves to Blackler's 'types of knowledge' as embrained, embodied, encultured, embedded, encoded, Tsoukas' 'processes of knowing' deriving from expectations, dispositions, interactions, situations, Nonaka’s ’knowledge conversion’ (socialization, externalization, internalization, combination) and Lam’s ’sharing knowledge across cultures’. But Walsham does not make the link explicit.

Most of these relate to human knowing of the kind Dooyeweerd’s discussed, rather than what emerges from information. But the concept that data, information and knowledge are ’in’ a data warehouse leads makes it easy to assume that we may treat data, information and knowledge as kinds of substance that are transformed into each other. Therein lies the problem.

In rejecting any substance-concept as an account of the nature of things, Dooyeweerd might clarify the issues. After data (analytic aspect) and information (formative aspect) should come not ’knowledge’ but ’signification’ (lingual aspect). It is signification that may be said to be ’in’ the data warehouse, that is shared as organisations ’share knowledge’, ’create knowledge’, etc. Unfortunately, the term ’knowledge’ has now stuck as the word referring to this, including in Newell’s ’knowledge level’ below.

*5.3.1.2 Long-term digital preservation*

This multi-aspectual understanding of information is of more than speculative interest. It becomes important, for example, in long-term preservation of information on computers. Dollar [2000:58] distinguishes "logical and physical structure, intellectual content, and context that were apparent at the time of creation or receipt", clearly indicating different aspects (in this case, despite the inexact use of ’physical’, they are, respectively, the analytic, psychic, lingual and social). But the digital preservation community has not yet agreed a set of such aspects, and this Dooyeweerdian view could help bring agreement and also separate out issues in research.

5.3.1.3 Virtual beings

What is the ontic status of beings we encounter in virtual world such as games like ZAngband (chapter 4), MUDs (Multi-User Dungeon games) or MMORPGs (Multi-Media Online Role Playing Games)? How is it possible that MMORPG players buy and sell pieces of virtual equipment in the real world (on eBay)?

The answer that these are ’only information’ (or a reality of ’pure mind’ that is otherwise equivalent to our own reality, discussed later), while it might be satisfactory as a theory, does not accord with our everyday experience of them. Is this a case where everyday experience must bow to theory, or is there a more satisfactory way to understand these phenomena that is philosophically sound?

Slightly better is Dooyeweerd’s discussion of the creative imagination of the artist [1984,III,p.113-116]. That which the artist imagines, such as the aesthetic idea of a beautiful human body sculpted in marble, is an intentional object of fantasy. He expands on this in [1984,II,p.387ff.]. Depicted in Fig. 5.3.1.3, the person who is functioning as subject in the analytic-logical aspect has a thought (intentional logical concept) and this might or might not be a logical object (thought-about denizen of the subject side). Virtual being or characters, according to this picture, have no logical object.

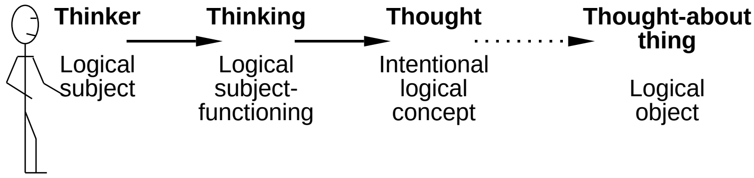


Figure 5.3.1.3.1. Dooyeweerd's understanding of thoughts

But it is unsatisfactory as an expression of everyday experience of virtual worlds, in which the virtual beings are not mere passive things like the intentional idea of the human body, but are highly active, in some cases, exhibiting many aspects of human living. Pacé Dooyeweerd, this author finds it more useful to conceive of virtual characters and items as actually existing in the subject-side cosmos as a range of aspects from the psychic onwards, as shown in Table 5.3.1.3.2. (Such a table could be useful in design of virtual characters, to ensure they are fully-rounded.)

Table 5.3.1.3.2. Aspectual beings of virtual character

| Aspect | The virtual beings ... |
|---|---|
| Psychic | appear on screen and make sounds |
| Analytic | are distinct from surroundings (or not) |
| Formative | plan and act intelligently (or not) |
| Lingual | communicate |
| Social | threaten and challenge (or not) |
| Economic | haggle over prices, are frugal and efficienc (or not, often humourously) |
| Aesthetic | (deliberately) do fun or funny things |
| Juridical | do virtual justice or injustice |
| Ethical | are generous or mean |
| Pistic | are committed to a deity |

But their existence is purely as object-functioning and never subject-functioning. It is programmed in the computer. Their existence in pre-psychic aspects is ignored, but if one must consider them, one can revert to the aspects of the hardware of the computer.

This view can be fitted into Dooyeweerd's, but is much more fruitful than his for understanding the important issues in virtual worlds. It is certainly more useful than the non-Dooyeweerdian belief that they are 'only information' or are living beings in a reality of pure mind.

### 5.3.2 Program and Software

What is a program? It might be true, but it is not sufficient, to say that it is a statement in computer language of what we want the computer to do. Consider the following:

> "I have an idea of a program I want to write and ponder it. I put those ideas down on paper. I write the program to implement those ideas and get it working properly. I store it on disk, and print in on paper. I run it, supplying whatever input is asks for. I run it again with different input. I give a copy of it to a friend and they run it."

How do we understand this? What is the difference between each and how do they relate to each other? Such questions, seldom considered, are important in, for example, legal cases when delineating various rights. A related issue concerns archives, especially digital archives. What happens, for example, in 500 years' time, when the coding between bits and symbols is lost and cannot be guessed at? What is the nature of such archives?

Since Dooyeweerd himself did not have much experience of computers, and he did not discuss such issues directly, we must apply his ideas to understand the nature of computer program, and will do so in two ways.

#### *5.3.2.1 Program as law side*

A program may be seen as a law side for a virtual world, which enables that virtual world to be and occur. The virtual entity side is the program actually running. This parallel is especially clear when the virtual world is that of a computer game or virtual reality, but it is also valid when the 'virtual world' is merely a few bits of data that model something.

- # Both Law side and programs enable being and occurrence.
- # Both entity side and running program is what exists and occurs.
- # Law side and programs are universalia while both entity side and running-program are specific instantiations of this.
- # Both law side and programs define what is meaningful.
- # Both law side and programs comprise distinct aspects of meaning-law.

Self-modifying programs (LISP programs or neural nets might be cited as instances) make this parallel difficult, since our cosmic law side does not get modified by the entity side, until we realise that program self-modification is always itself enabled by the program. But this is an open question that is still to be addressed.

The parallel between program and law side could be used to test the shape of Dooyeweerd's notions of law and entity sides. It would not test its validity, which is a pre-theoretical stance, but it could help test and refine our understanding of its shape, in the vein of Dennett [1998] when he suggests that AI can provide new ways to test philosophy in general. We might construct special programs to undertake such testing, but this might not be necessary because a survey, examination and exploration of a wide variety of existing programs would yield many insights. If it is objected that programs are built of propositional statements, which can never fully express meaning, we must remember that program statements do not have to be propositions, but often include non-propositional elements like neural nets and spatial or fuzzy constructs (see below). However, it should always be remembered that programs are always to be understood in terms of human subject-functioning, which is a response to the 'real' law side, and so the philosophical relationship between this and the pseudo law side that is program should be kept in mind.

### *5.3.2.2 Program as performance art*

However a rather more penetrating analysis of the manifestations of program might follow Dooyeweerd's analysis of art. In discussing Praxiteles' sculpture of Hermes and Dionysus, he said [Dooyeweerd, 1984,III,p.116-117] (remember that Dooyeweerd's use of 'objective' and 'subjective' has nothing to do with whether something is fact or opinion):

> "First, observe that the vital [organic] function of Hermes and the boy Dionysis was objectively intended in the artistic conception of these figures. ... the artist indeed had a productive vision of two living deified human bodies. The organic vital function of Hermes and the boy Dionysus was thus undoubtedly implicitly intended in his productive fantasy. This aesthetic intention is realized in the objective structure of the statue, as a thing. ... in its aesthetic structure, the intentional vital function has been objectively represented or depicted. And this objective representation belongs to the reality of the marble ..."

In this we see the relation between what creator intended and its representation in some medium, and its reality in that medium. A computer program is not fine art, but the programmer's idea is likewise represented in some program and gains a reality in that medium. But a program is more like performance art, of which Dooyeweerd said [Dooyeweerd, 1984,III,p.110]:

> "It would be incorrect to assume that all works of fine art display the structure of objective things. This will be obvious if we compare plastic types (i.e. painting, sculpture, wood carvings, etc.) with music, poetry and drama.
>
> "Works of art belonging to the last category lack the constant actual existence proper to things in the narrower sense. They can only become constantly objectified in the structure of scores, books, etc. .. such things as scores and books, are, as such, symbolically qualified. They can only signify the aesthetic structure of a work of art in an objective way and cannot actualize it.
>
> "This is why artistic works of these types are always in need of a subjective actualization lacking the objective constancy essential to works of plastic art. Because of this state of affairs they give rise to a separate kind of art, viz. that of performance, in which aesthetic objectification and actualization, though bound to the spirit and style of the work, remain in direct contact with the re-creating individual conception of the performance artist. The latter's conception, as such, cannot actualize itself in a constant form, though modern technical skill has succeeded in reproducing musical sound-

waves by means of a phonograph."

Considering these two quotations together, we may draw the following parallels:

- # There is an intentional object, which is the thought-up idea which the creator wants to express, and does so in the medium.
- # The program (written down: written-program) itself is like a music score, both of which may be printed on paper or stored on disk.
- # Both are symbolically (lingually) qualified.
- # Both written-program and music score are symbolic significations of what the creator intended, in a chosen language of notation, which differs from natural language. The language might use text as a medium, such as C or Java, or might use graphics, like music scores or visual programming languages.
- # The playing of the music is like the running of the program in the computer (running-program); this results in a actualization of both music and program.
- # Just as the score is not the music, so the written program is not the running-program.
- # Just as the score persists while the performance is transient, so the written-program persists while the execution of running-program is transient.
- # It is the human performance artist who plays the music and the human user who runs the running-program, and supplies input.
- # One performance of a piece of music differs from the next; one run of a program differs from the next, usually in that different input is supplied but even when the same input is supplied there are other differences, such as location in memory.
- # The score and program both express a general ability to perform in a range of different situations, in one the ability to perform this music, in the other the ability to execute the program.
- # This allows someone else to play the music, run the program.
- # The recording of a performance equates with a recording of the running of a program (by means, for example, of a keystroke recorder or a video of the screen.) What the I.S. developer creates is like what the composer creates: a symbolic signification of what s/he intends.

This gives us a foundation for understanding the different manifestations of program exemplified above.

It also indicates the point of contact with program as virtual law side. It is clearer if we consider a game played rather than a piece of music (both qualified by aesthetic aspect). A game has rules that make the concrete playing of it possible and directs it without determining its outcome. The playing of the game, the performing of the music and the running of the program positivises their laws, just as the actual being and occurring of the cosmos positivises the cosmic law side.

Note that the process of programming usually involves running a program development system (PDS), which might comprise, for example, a text editor and a compiler. Therefore, programming is seen as an application task and the programmer is seen as user of the PDS (but not of the program s/he is writing) in the manner of chapter 4. This fascinating process of programming, in its wider context of knowledge elicitation and system development, will be the topic we seek to understand in chapter 6.

## 5.4 NEWELL'S LEVELLED UNDERSTANDING OF COMPUTERS

We now consider a number of ways in which the nature of computers has been understood as a framework for research and practice, and one attempt at a comprehensive framework for understanding.

### 5.4.1 Ways of Understanding Computers

Perhaps the first way by which the nature of computers was understood was as electronic hardware, whether digital or analog. While computers may still be seen as such (especially to allow for analog computers), this way ceased to be dominant in the 1960s, when it was replaced by a focus on digital bits held in memory and manipulated by a central processor unit (CPU). At the time, the program was seen in terms of machine code instructions, originally set manually by pushing buttons (one per bit on the Honeywell DDP-516, which the author used), but soon made more convenient by means of assembly and autocode languages. Within this framework for understanding arose the Turing and Von Neumann machine models.

By the 1970s, this framework for understanding computers was no longer dominant, having been replaced by one based on symbols and what were then called 'high level languages' and later called 'third generation languages'. Within this framework arose many models, most of which we discuss in chapter 7, including the entity-relationship, relational and object-oriented data models, various knowledge representation approaches such as procedural, functional, logic and object-oriented programming.

During the 1980s, the focus in business shifted to what the symbols stood for, namely the application, and in AI the agent perspective took over. These share a framework for understanding computers based on the application content of what is represented in the program, together with such things as the agent's goals, purposes or preferences.

Each of these frameworks for understanding are still valid, and can work together.

From our Dooyeweerdian understanding of the computer as a multi-aspectual meaningful whole, we can immediately see that each of these extant ways of understanding the computer is centred on a different aspect:

- # Computer as hardware: organic aspect
- # Computer as bit-manipulation machine: psychic aspect
- # Computer as symbols: analytic and formative aspects
- # Computer as agent or content: lingual aspect.

However, we will not examine them further individually here because there is a comprehensive framework for understanding the nature of computers which integrates them all into a single framework. It is this we shall discuss in detail.

### 5.4.2 Newell's Levels

Allen Newell, an eminent figure in early AI, did what Dooyeweerd suggested had not been done: analyse the structure of a thing, the computer, from the point of view of naïve experience. In his classic, ground-breaking paper, 'The Knowledge Level' [1982], he brought together several ways of understanding a computer, into one system of levels. He claimed that his levels are not derived from a priori theory but derived primarily by reflection on years of practice in artificial intelligence [1982,p.92]. This is perhaps why his proposal of levels has been so useful not only within AI but also within the HCI community. 'The knowledge level' is his most-cited paper.

His main concern was to explore the intuitive distinction between knowledge and the symbols that hold it and to understand "What is knowledge?", "How is it related to representation?" and "What is it that a system has when it has knowledge?", and he presupposed that the answers to these questions can be the same for both human beings and computers. His paper addressed two main issues: the levelled nature of the computer, and why it is that at one level behaviour of an agent is deterministic while at another level it is not.

To Newell, any computer system can be described at several distinct levels:

- # Device level, whose medium is electrons and magnetic domains in physical materials: looking at hardware as physical materials ('device' refers to such as semiconductor P-N junctions)
- # Circuit level, whose medium is voltages and currents in electronic components: the view of hardware we discussed above
- # Logic ('bit') level, whose medium is bits in computer memory and registers: bit-manipulation machine ('logic' refers to digital bits, not to reasoning)
- # Symbol level, whose medium is symbols in data structures: computer as symbolic program
- # Knowledge level, whose medium is knowledge: what the symbols are about, "aboutness": computer as agent.

Each level is a way of seeing the computer, and "Neither of these

.. definitions of a level is the more fundamental. It is essential that they both exist and agree." (ibid.,p.95) Different levels describe the same system, not different parts thereof, and do so in equally valid ways, so a description at a level is complete, in the sense of not leaving gaps that must be filled in by reference to descriptions from other levels.

Newell worked out his notion of levels to some detail. Each level provides a set of concepts and vocabulary for describing a system that includes (Newell, 1982,p.95) "a medium that is to be processed, components that provide primitive processing, laws of composition that permit components to be assembled into systems, and laws of behavior that determine how system behavior depends on the component behavior and the structure of the system".

"It is noteworthy how radically the levels differ. The medium changes from electrons and magnetic domains .. to current and voltage .. to bits .. to symbolic expressions" .. to knowledge. On the other hand, "some intricate relations exist between and within levels."

Each level defines a distinct technology. If a system has a description at one level then it will always be possible to describe it at the next lower level. Because of this, lower level technologies are used to implement higher ones, and it will always be possible to realize any level's description as a physical system.

But the reverse is not always the case. "Computer systems levels," said Newell (1982,p.97), "are not simply levels of abstraction. That a computer has a description at a given level does not necessarily imply it has a description at higher levels." "Within each level," stated Newell [1982,p.95], "hierarchies [aggregations] are possible," but merely aggregating things at one level does not necessarily take us up to the next level -- though Newell never clarified exactly what it is that takes us up to the next level.

While much of the above has been discussed, most have overlooked a curious claim Newell repeated for his suite of levels:

> "They [levels] are not just a point of view that exists solely in the eye of the beholder. This reality comes from computer system levels being genuine specializations, rather than being just abstractions that can be applied uniformly." (ibid.,p.98)
>
> "Nature has a say in whether a technology [and therefore a level] can exist." (ibid.,p.97)
>
> "To repeat the final remark of the prior section: Computer system levels really exist, as much as anything exists. They are not just a point of view. Thus, to claim that the knowledge level exists is to make a scientific claim, which can range from dead wrong to slightly askew, in the manner of all scientific claims." (ibid.,p.99)

This is a strong ontological claim. But, to Newell's regret [1993,p.33] "no one has taken seriously -- or even been intrigued with -- the proposition that the knowledge level was not invented", and to my knowledge this is still true.

Following this, Newell addressed the problem of non-determinate knowledge level behaviour, which we discuss below.

Newell did not analyse his notion of levels philosophically. Towards the end of [1982] Newell he suggested the knowledge level resembles Dennett's [1978] intentional stance, but there are significant differences from Dennett (such as in the number of levels/stances, and in Newell's ontological claim), and Newell called for closer analysis [1982,p.123].

A philosophical analysis is certainly needed because ontic irreducibility between levels, stances or technologies is a philosophical rather than a scientific issue. This might explain the rather mixed reception the notion of the knowledge level received that Newell reported in [1993,p.34-36] by six communities. One embraced the notion because it gave them "a way of talking about what knowledge a system must have without regard to how it is to be represented", but did so uncritically. One adopted it as a framework to stimulate a new way of looking at learning. Three communities, to which it is potentially relevant, largely ignored it, not because they disagreed with it but because they were not interested in philosophical matters like defining intelligence or they were happy with current philosophical underpinnings. The only community that made it central was the SOAR community, in which Newell himself was closely involved.

It is difficult, however, to find a philosophy that critique Newell's proposal immanently, doing justice to it and respecting rather than dismissing his ontological claim. Much philosophy of the past century, including Dennett's, ignores ontology, and that which does not is usually reductionist in flavour and thus fundamentally incapable of allowing for a plurality of levels. Dooyeweerd's philosophy, however, might allow us to undertake a philosophical analysis of Newell's proposal.

### 5.4.3 A Philosophical Analysis of Newell's Proposal for Levels

We can immediately detect a similarity between some of Dooyeweerd's aspects and Newell's levels, shown in Table 5.4.3.1 with reasons which are elaborated later.

The similarity is strengthened when we compare the two notions of levels and aspects as such:

- Both provide different ways of describing the same thing.
- The levels and aspects occur in the same sequence.
- Both levels and aspects exhibit irreducibility of meaning.
- Both levels and aspects exbibit inter-aspect/-level dependency.
- What Newell calls system might be Dooyeweerd's enkaptic structural whole.
- Levels, like aspects, involve laws.
- Just as each level defines a distinct technology, so each aspect defines a distinct area of science.
- Newell stated [1982,p.95] "Within each level hierarchies are

possible" just as within each aspect aggregation occurs.

# Newell's exploration of the knowledge level might be seen as his exploration of the lingual aspect as it relates to computers.

# That knowledge is generative reflects its formative aspect (on which the lingual depends).

Table 5.4.3.1. Comparing Newell's levels and Dooyeweerd's aspects

| Aspect | Level | Reason |
|---|---|---|
| Quantitative<br>Spatial<br>Kinematic | | |
| **Physical** | **Device** | Enerty, electrons |
| **Organic-Biotic** | **Circuit** | Distinction from environment |
| **Psychic-Sensitive** | **Logic 'Bit'** | Signals, states |
| **Analytic** | **Symbol 1** | Decision, distinction |
| **Formative** | **Symbol 2 (PSCM)** | Processing, structure |
| **Lingual** | **Knowledge** | 'Aboutness' |
| Social<br>Economic<br>Aesthetic<br>Juridical<br>Ethical<br>Pistic | | |

Further, four similarities may be found in the two approaches.

# Newell [1982,p.123] spoke of the relationship between symbol and what it stands for as 'aboutness', but "the knowledge level does not itself explain the notion of aboutness; rather, it assumes it." This is reminiscent of the kernel meaning of the lingual aspect (signification) being graspable not by theoretical thought but only by intuition (§3.1.4). We may see Newell's detailed exploration of the relationship between knowledge level and symbol level as an exploration of the lingual aspect as it relates to computers, which might contribute to Dooyeweerdian scholarship.

# Newell claimed that his levels are not derived from a priori theory but derived primarily from years of practice in artificial intelligence [ibid.,p.92]; Dooyeweerd's aspects are derived from years of reflection on everyday life (§3.1.6).

# Newell made a strong ontological claim for his suite of levels (though he recognised that this claim could "range from dead wrong to slightly askew, in the manner of all scientific claims" [ibid.,p.99]. Likewise, Dooyeweerd made a similar, though subtly different, claim, that the aspects are not just a point of view, though his suite of aspects should be subject to criticism and refinement (Dooyeweerd, 1984, II,p.556). The subtle difference is that the aspects cannot be said to 'exist' so much as 'pertain', since they are the very framework that

makes existence possible (§3.1.3, §3.1.4).

There are however some differences between them, such as that Newell did not recognise what Dooyeweerd called anticipatory dependency. Nevertheless, we have good reason to propose that Newell's levels are remarkably similar to Dooyeweerd's aspects, and we can propose Dooyeweerd's philosophy as one to provide a philosophical grounding for Newell's levels.

### 5.4.4 Level-aspect Correspondences

The correspondence between the individual levels and aspects shown in Table 5.4.2.1 may now be examined more closely. The device level, concerned with physical materials, is obviously the physical aspect of the computer. The knowledge level's concern with 'aboutness' closely matches the lingual aspect's 'signification'. But the other level-aspect correspondences are less obvious. We must avoid being misled by the labels used for levels or aspects and focus on what each thinker referred to when speaking of it.

That Newell's logic (bit) level corresponds with the psychic aspect becomes clearer when we consider alternatives. Briefly, the argument is as follows. The physical, biotic, formative or lingual aspects are ruled out because the bit, as a digital state of being on or off or of switching between them (a 'signal') has no meaning within them, which leaves the psychic or analytic. The psychic aspect seems more appropriate when we treat such states as object-functioning as part of human (or even animal) experience: the colours on screen are aggregations of pixels each of which holds a state. Finally, this is supported by a strong similarity between things at a level of a computer system and aspects of a human (or animal) subject. The psychic, as opposed to analytic, operation of human memory and recognition (Dooyeweerd drew attention to animals' distinctions of e.g. mates [Dooyeweerd, 1984,I,p.39]), involves the activation states of neurones without logical functioning.

What Newell [1982] identified a single symbol level corresponds with two aspects, the analytic aspect of distinction and conceptualization, in which basic types of data -- integer, boolean, text, etc. -- are meaningful, and the formative aspect of deliberate shaping, in which data structures and algorithms are meaningful. However Newell later said [1993,p.36-37, our italics]:

> "Gradually it has become apparent that between the knowledge-level description of a Soar system and the symbol-level description (the one that talks about recognition memory, working memory, the decide process, impasses, and chunking) there is an organization in terms of problem spaces, which in many ways is like another computational model. One need only talk about operators, states, desired states, selection knowledge for operators, etc. This must be a symbol-level organization (we know of no way to have a genuine system level between the symbol level and the knowledge level), but different from ... the recognition memory etc."

From the text we have italicised, it is clear that Newell himself had grasped intuitively the distinction in kernel meaning of Dooyeweerd's analytic and formative aspects. But while Newell felt constrained by his ontological commitment to a single symbol level made a decade earlier, Dooyeweerd does offer a "way to have a genuine system level between the symbol level and the knowledge level".

Finally, it seems strange to align the biotic aspect, usually seen as having a kernel meaning of life functions, with an electronic circuit level. There are several reasons for this, which have been discussed earlier, the main one being that one important way in which physical meaning is inappropriate to both a living organism and a manufactured circuit or piece of hardware is that both are distinct from their environment whereas in the physical aspect, fields and forces pervade all.

### 5.4.5 Enriching Newell's Notion of Levels

If Dooyeweerd can provide a philosophical foundation for levels, then we have a basis for not only affirming Newell's notion of levels, as we have demonstrated above, but also critically enriching it. For example:

# We have already noted the possibility of enriching the symbol level by reference to two aspects.

# While Newell held that each level provides a distinct description and has distinct types of laws of composition and behaviour, Dooyeweerd's aspects involve inherent normativity; so could this provide Newell's levels with the ability to give normative guidance to those working at each level?

# Because Dooyeweerd's suite contains more aspects than Newell's set of levels, this suggests that there might be other levels above the knowledge level -- and indeed, Jennings [2000] has argued for a social level.

# Though Newell, working within a positivist tradition, did not mention meaning, it seems, to this reader at least, that he was reaching for meaning, in Dooyeweerd's sense of 'referring beyond'.

Dennett [1998,p.284] criticised Newell's lack of clarity about 'aboutness'. To Newell, the role of a symbol is to give 'distal access' to knowledge, and he depicted this in [1982] as an arrow from the symbol to what it 'accesses'. This might be further knowledge in the agent, but, as Dennett pointed out: , "Those [distal access] arrows ... lead one either to more data structures or eventually to something in the external world -- but he is close to silent about this final anchoring step. ... Newell sweeps [this issue] under the rug right here." However, Dooyeweerd might come to Newell's rescue in explaining why Newell swept 'aboutness' under the rug: it concerns the kernel meaning of the lingual aspect, symbolic signification, which cannot be grasped by theoretical thought.

In short, Dooyeweerd seems to be what Newell was reaching for in his theory of levels. By reflection on what AI researchers and developers had come up with, Newell wanted to see both computers and human beings through a lens of multiple levels that are ontically, irreducibly distinct ways of describing this, each of which has a distinct set of concepts, medium, laws, technology, etc. and in which both deterministic and non-deterministic behaviour can be incorporated. (We might note also that systems thinkers, such as Bunge and Hartmann, are also reaching for level plurality, though they do not work it out in the way Newell did; see later.) This is precisely what Dooyeweerd offers.

### 5.4.6 Some Practical Implications of Aspectual Levels

It might appear from the above that Newell's levels were selected as a framework to analyse using Dooyeweerd simply because it seemed promising as a recipient of Dooyeweerdian attention. In fact, the author first discovered Newell's levels in the early 1980s, before returning to academic life, and long before he discovered Dooyeweerd. See Vignette 4 in tne Preface. Immediately they appealed to him because they accounted for what he was experiencing in information systems at the time. When he returned to academic life, he used the levels to structure my undergraduate and postgraduate teaching in a number of modules in order to ensure that he covered a wide range of relevant issues in a way that does not confuse them, and because it imparted the 'wisdom' that integrates the human and ethical with the technical. He still does. Table 5.4.6 shows how he structured various courses according to the levels.

Table 5.4.6. Structure of Courses

| Level (aspect) | Database | Multimedia (+ web sites) | User Interface | Psychology in HCI |
|---|---|---|---|---|
| Device/ Materials (physical) | Magnetic v optical tgy | - | - | Chemistry of nerves |
| Circuit (organic) | Disk electronics, mechanics | MM displays, sound systems | UI devices | Nervous system, ears, eyes, muscles |
| Logic/bit (psychic) | Disk tracks, sectors, checksums Data security | As UI; Anti-aliasing, Rendering Animation speeds Sound, Lip Sync | Graphics, sound samples, fonts, Windows, Gestures | Memory, Pattern detection, recognition; Stim+Resp; Behaviorist psy |
| Symbol 1 (analytic) | Basic data types, fields, Indexing, Transaction processing | Basic types of info to show | Types of info to show | Concepts |
| Symbol 2 (formative) | Records, Data models Keys | Structure: Page, links, Animation paths, 3D models | Structure: complex info | Semantic, procedural memory, Attention, Cognitive psy |
| Knowledge (lngual) | Normalisation, Content, Knowledge management | The story; Accuracy, etc. | The content | Human meaning and behaviour; Psycho-analysis |
| Tacit (Social) | Cultural connotation | The feel | Nuances | Social psychology |

(Such tables could be used more generally in course design, covering all aspects.) Note that a level was added above the knowledge level, so far called the tacit level, the main theme of which is cultural connotations -- and thus is social in nature. Gradually, his use of these is taking on a more Dooyeweerdian flavour, with separation of the symbol level into analytic and formative aspects.

## 5.5 COMPUTERS AND HUMAN BEINGS

An issue that has taxed us since computers arrived is the similarity and difference between computers and human beings -- can computer think, understand, etc.? -- which is the central question of artificial intelligence. Boden [1990] has made a collection of key papers on this issue, but it shows that while the debate continues, it does not seem any nearer a resolution than at the start. To address the AI question from a Dooyeweerdian standpoint we must first establish the basis for debate and nature of such a comparison. Should it be focused on the material-mental (brain-mind) dichotomy (McCulloch and Pitts), or on the contrast between the deterministic machine and the non-determinacy of human free action (Newell), or on whether machines could ever truly understand (Searle), or on Dennett's broader notions of intentionality, or on Turing's test based on how questions are answered? It is not the intention to attempt to make a substantive contribution to the debate here, let alone a comprehensive resolution, but rather indicate what might be a new, fruitful way forward towards such a contribution. We will examine two of the bases for debate, Newell's attempt to account for non-determinacy of knowledge level behaviour while assuming that computers and humans are essentially the same, and Searle's thought-experiment of the Chinese Room, by which he hoped to demonstrate the opposite.

### 5.5.1 Determined and Non-Determined Behaviour

In the second half of his [1982] paper, which have already discussed, Newell noted that knowledge level behaviour is unpredictable and undetermined (for example, he cited Frank Stockton's [1895] story The Lady or the Tiger) but symbol level behaviour is predictable and determined (because it consists of mechanical processing of symbols). Why is determinacy suddenly lost between the symbol and knowledge levels?

Newell tried to account for this by defining knowledge as the logical closure of, i.e. what could ever be deduced from, all that is represented in the agent's mind. In brief, his argument proceeds: Since this is infinite, in any concrete decision-situation, the agent must make use of only part of this 'knowledge', so we can only predict what part this will be by looking at exactly what symbolic representations the agent has -- which means that to understand the behaviour of the agent at the knowledge level, a knowledge level description of the agent is 'radically incomplete' and must be augmented with some symbol level description.

But this argument proved "rather hard to understand" [Newell, 1993,p.33], and his definition of knowledge as logical closure is so completely at variance with everyday experience of what knowledge is that its veracity must be questioned. Also, why should the knowledge level be unlike other levels of description in being ’incomplete’, other than that to believe so is required by Newell’s argument?

Furthermore, Newell restricted his attention to the ability of an observer to predict the agent’s behaviour, and completely ignored the issue of whether behaviour can be non-determined as such. So it does not really address the AI question. We might also note that his initial assumption that knowledge level behaviour is non-determined while symbol level behaviour is determined might itself be questioned, because the only direct experience of non-determined knowledge level behaviour is in human beings and our only experience of determined symbol level behaviour is of computers (since we have no unambiguous access to what we might think of as symbol structures in the human mind but do have access to these in computers because we programmed them).

Thus Newell’s attempt to discuss the AI question in terms of determinacy leaves many questions unanswered.

The root of the problem lies in the presuppositions of the community of thought in which he was working, which are brought into sharp relief by reference to Dooyeweerd.

- # The very formulation of the problem, that determinacy and non-determinacy require explanation and are fundamentally incompatible presupposes the nature-freedom ground-motive.

- # That Newell felt compelled to explain why non-deterministic behaviour emerges from determined behaviour at the level below arises from the presupposition of self-dependent substance, on which all else depends. In this case, the ’substance’ is deterministic in nature (whether it be symbols or, as suggested by the first part of the paper, physical, does not matter here).

- # That Newell believed knowledge level behaviour to be non-determined and symbol level to be determined arises from the AI presupposition that computers are, in principle, completely equivalent to human beings at the symbol and knowledge levels. It is by the presupposition that he assumed that symbol level behaviour is determined and predictable while knowledge level behaviour is not.

### 5.5.2 The Chinese Room

Searle [1990] crystallised the debate by proposing a thought experiment to demonstrate that the claims of what he called ’strong AI’, that appropriately programmed computers can genuinely understand (possess intentionality), and that programs thereby explain the human understanding, are baseless. To summarise: Suppose I do not understand Chinese, and cannot even recognise Chinese writing from any other shapes. I am in a room with a batch of such Chinese writing. From time to time more pieces arrive through a hole in the wall. I also have a rule book in English (which I understand well) that tells me how to reply (by drawing shapes) to each received pattern on the basis of formal properties like its shape and which take into account all previous patterns received and sent. Also, occasionally, I receive questions in English, and reply to those. "Where," asks Searle rhetorically, "in this room is the understanding of Chinese? And how does it differ from my understanding of English?" He argued that computers running a program are like my following the rule book and cannot understand in the way human beings do, and that programs are not even necessarily part of our understanding and hence do not constitute a useful explanation:

> "the programmed computer does not do 'information processing'. Rather, what it does is manipulate formal symbols. ... The computer .. has a syntax but no semantics." [p.85]

He argues that biological causality is necessary for understanding, and that physical causality can never achieve this; humans operate by one while computers operate by the other.

### 5.5.3 The Debate

Various counter-arguments have been attempted by AI supporters, including

- # the systems reply, that while I do not understand Chinese, the system of room, rules, me, etc. as a whole does, as an emergent property,
- # the robot reply, that understanding involves action in the world so the Chinese Room would understand if only the symbols I draw in reply are sent to robotic arms etc.,
- # the brain stimulator reply, that all we need is for the program to simulate the operation of brain cells rather than rules directly,
- # the combination reply, that putting all of these together is enough for genuine understanding,
- # the other minds reply, that we cannot know what is in another mind except by the behaviour we see, so if the Chinese Room behaves aright we may say it understands Chinese, and
- # the many mansions reply, that eventually we will build computers with the right type of causality and these will truly understand.

Searle has countered all of them successfully [Searle, 1990,p.72ff].

Boden [1990] gives more substantial arguments against Searle's view. First, she suggests that Searle's argument involves a category mistake, in that it is not the brain (me in the Room) that understands, but the person -- and in this Dooyeweerd would agree because it is the meaningful whole, not the part, that functions; see §3.2.6.2).

But her next two arguments are weaker. Second, following the rule book does involve understanding -- an understanding of English in which the rules are written. She seems to miss the obvious rejoinder, that understanding the English in which the rules are written still does not enable understanding of Chinese, which is what Searle was concerned with.

Third, she argues against Searle's statement that computer has syntax but no semantics by arguing that for the internal rule-following program to run there must be some procedural element, and that this constitutes semantics. But her attempt to locate semantics in the Room in the procedural following of rules is the wrong semantics: Searle is concerned with the semantics of Chinese, not of rule-following.

So Searle's question, "Where is the understanding of Chinese?" seems to remain unanswered. We have two ideologically-motivated camps, each of which simply rejects the arguments of the other, and between which there is no real communication. Nothing has been resolved, and the debate has not even thrown much light on the issues.

### 5.5.4 A Critique of the Debate

Before an answer to Searle's question is attempted, Dooyeweerd would urge us to examine the debate itself, especially its presuppositions that both sides adopt without question. His different starting point for philosophy can be used to expose some of these.

First, we might detect a version of the Nature-Grace ground-motive (NGGM), in which human beings are 'sacred' and computers, 'secular'. When a debate occurs between supporters of two opposing poles of a dualistic ground-motive no logical resolution is possible because it is the nature of ground-motives that they are religious in nature (in the sense defined by Dooyeweerd) and this involves tenaciously held commitments, in defence of which reason is harnessed. Resolution involves shifting to a different ground-motive, which is attempted below.

Second, and commensurate with the NGGM, we see in the debate about what types of causality is necessary for true understanding and intentionality, a presupposing of a substance-idea. That is, both proponents and opponents presuppose that there is some 'substance' (in the Aristotelian sense of fundamental principle on which all else depends, and in this case the substance is not a static 'stuff' but a type of causality), and their contention is about what this substance is that is necessary for understanding. The debate centres on what a computer is and can do 'in and of itself'.

One problem with substance-presuppositions is that we end up with dogmas for which little justification can be offered. Two of these can be seen in Searle.

- # He claims that "the programmed computer does not do 'information processing'. Rather, what it does is manipulate formal symbols. ... The computer .. has a syntax but no semantics" [p.85]. But on what basis may we reject information processing or semantics while accepting symbol-manipulation and syntax? Indeed, on what basis may we reject or accept either of these on their own? As has already been noted, if we open up the case of a computer we find no symbols there being manipulated!

# He claims that biological, not physical, causality is necessary for 'information processing' and understanding, and that the former cannot be reduced to the latter. Yet Searle does not justify this belief, nor does he explain why it is that biotic causality can support understanding yet physical causality cannot. He holds this as a dogma.

His opponents hold equivalent, reverse dogmas.

### 5.5.5 Towards a Resolution

Both Searle and Newell work within the substance presupposition. Dooyeweerd would urge us to critically examine all these presuppositions, and might offer the following counter presuppositions:

# The creation-fall-redemption ground motive accepts both determinacy and non-determinacy, though in different aspects of the same thing. It refuses to allow a divorce between sacred and secular.

# That existence, and the structure of things, may be derived from meaning, provides a way of allowing for multi-aspectual behaviours without having to see one as emerging from another.

# The distinction between meaningful- and subject-functioning, which we introduced earlier, allows us to see computers and humans as alike in one way but not in another. Both symbol level and knowledge level behaviour are human subject-functioning, but both computers and humans can function meaningfully in both. (Note that to Dooyeweerd, symbol level functioning is also non-determinate, because it is analytic and formative functioning that interprets the physical operation of the computer.)

If we do not wish to adopt all these presuppositions, we might note that they help us in different ways. The third alternative leads us to question certain assumptions. The second removes the need to explain by emergence. The first dissolves the problem.

However, there is a secondary level to the debate, which is not so bound by the NGGM, namely the recognition that it is possible there are two distinct types of causality. This could, of course, result in dualism. But it might alternatively be seen as a subset (two in size) of the full set of aspects, since each aspect defines a distinct type of causality or repercussion (as was elaborated in §3.1.5). If we define understanding (or information processing) in terms of post-biotic functioning, then, owing to inter-aspect dependency it will necessarily involve biotic functioning, affirming Searle's view, but not in the way he would expect. It also affirms his opponents' views that physical causality is also necessary (though both he and they conflate the physical with the analytic aspects).

In the same way, as has already been explained, syntax and symbol manipulation may be seen as of the formative aspect while semantics and what Searle calls information processing may be seen as functioning in the lingual aspect.

What Searle holds as dogma -- the distinction between biotic and physical, and that between semantics and syntax -- is revealed to be an outcome of, and groundable in, Dooyeweerd's wider theory of aspects. But the way both humans and computers function in these aspects becomes clearer when we apply Dooyeweerd's non-Cartesian notion of subject and object.

### 5.5.6 Subject- and Object-Functioning

As explained in §2.4.5. §3.1.5, an entity can function in an aspect as either object or subject (agent). If it functions as object, then it does so as part of some other agent's subject-functioning. Everything can function as object in any aspect. But humans can function as subject in all aspects, animals can function as subject only as far as the sensitive aspect (though possibly higher primates might extend into the next couple of aspects), plants, as far as the biotic, and non-living things only as far as the physical.

Since the computer is not living, the latest aspect in which computer functions as subject is the physical. That is, it can function 'on its own' and without human functioning only as far as the physical aspect. As explained above, the computer's object-functioning in later aspects is meaningful only by virtue of our ascription of those aspects' meaning to its physical subject-functioning.

If we think only in terms of subject-functioning then we cannot validly say such things as "The computer thinks." But if we think in terms of its object-functioning then we can do so. And doing so is neither metaphor nor anthropomorphism.

The view here is echoed in Milewski's [1997] 'delegation' proposal, that we should understand agents not in terms of their innate characteristics but in terms of the relation they have with users. But our view extends to any human, including developers too, and it also allows for subject-functioning in the physical aspect.

### 5.5.7 A Fresh Look into the Chinese Room

Dooyeweerd can now let us approach Searle's question, of where is the understanding of Chinese, in a different way than either Searle or his opponents do. We argued above that it is valid to say "The computer knows X", but only if we speak, not in terms of subject-functioning, but rather than what we might call meaningful-functioning, aspectual functioning as either subject or object. In terms of meaningful-functioning, we can say that genuine knowledge of Chinese is located in the rule book (after all, somebody wrote it, so it is an object of human understanding-functioning) -- an answer which, interestingly, nobody in the debate seems to have seriously considered, though Boden does get near it sometimes.

What Searle was at pains to argue against is what we have called subject-functioning, and we would agree with him. But what some of his opponents might include is the meaningful-functioning. With this differentiation, we are able to welcome contributions from both camps as insights into the whole issue. Their apparent incommensurability dissolves when we move to the Creation-Fall-Redemption ground-motive which Dooyeweerd presupposes, because it allows for a cohering diversity in which Meaning is central.

Searle did in fact allude to what we have called meaningful-functioning [1990,p.72]:

> "We often attribute 'understanding' and other cognitive predicates by metaphor and analogy to cars, adding machines, and other artefacts, but nothing is proved by such attributions. We say 'The door knows when to open because of its photoelectric cell,' 'The adding machine knows how (understands how, is able) to do addition and subtraction but not division,' and 'The thermostat perceives changes in the temperature.'"

He quickly dismissed this approach with, "but I take it no philosophical ice is cut by such examples. ... the issue would not be worth discussing" and thereby failed to explore it and discover that it is by no means 'metaphorical' and is rich in terms of multi-aspectuality. His quick dismissal led him to overlook what might be the only way to resolve the question in a fruitful manner.

Searle's claim that the computer "has a syntax but no semantics" may now be examined. We agree that semantics is "solely in the minds of those who program them and those who use them" on the grounds that it is our (lingual) subject-functioning, which ascribes lingual meaning to the computer. But is not this true of syntax too? Syntax is formative meaning ascribed to the computer. The only thing the computer 'has' of itself without reference to us is its physical subject-fnctioning. All else is its object-functioning ascribed by us.

Table 5.5.1 summarizes this, comparing strong AI, Searle and Dooyeweerd in aspectual terms. In these terms, strong AI believes that computers can operate 'on their own' and without reference to humans (that is, function as subject) in all aspects from physical to lingual (and beyond). Searle believes that they can do so only as far as the formative aspect of syntax. But Dooyeweerd has two answers: in terms of subject-functioning the computer can only function as subject in the physical aspect, but in terms of subject- and object-functioning taken together (meaningful-functioning) it functions in all aspects.

To the meaningful-functioning question, the answer is "Yes!" So statements like "The Prospector program found a molybdenum deposit" are as meaningful and as valid as "Jim Smith found a molybdenum deposit using Prospector" -- so long as we see the first as M-F and the second as S-F. To do so is neither anthropomorphism nor a metaphor. In terms of meaning, computer and human are alike, because both function within the same meaning-framework, but they function in different ways, one as object, one as subject. Likewise, it is valid -- under lingual object-functioning -- to say that the computer 'knows', 'understands', 'has an intention towards', and the like.

Table 5.5.1 Views of Functioning of Computer

| Aspect | Dooyeweerd S-F | Dooyeweerd M-F | Searle | A. I. |
|---|---|---|---|---|
| Lingual (semantics) | No | Yes | No | Yes |
| Formative (structure) | No | Yes | Yes | Yes |
| Analytic (typed data) | No | Yes | Yes | Yes |
| Psychic (digital) | No | Yes | Yes | Yes |
| Physical (materials) | Yes | Yes | Yes | Yes |

The benefit of this approach is that the debate moves away from conflict based on dogma to seeing both sides as part of a wider picture, because it exposes their presuppositions to scrutiny and re-grounding in a different ground-motive.

### 5.5.8 Cyberspace and Bodiless Reality

John Perry Barlow's [1996] Declaration of the Independence of Cyberspace is a polemic that claims Cyberspace is a different type of reality, a reality of mind without body or matter. Cyberspace is a reality composed of information, thought, knowledge, and there is no need for a body or matter, or at no need for any particular body or matter. If this is so, he claims, then we should have different social arrangements, different ethics, different views of what is considered criminal or legal, different legislative frameworks and different freedoms. The legal systems of the old, matter-based reality no longer apply.

In direct opposition to Barlow is the feminist notion of embodied knowlege. Not only does knowledge need a body, but propositional, conceptual knowledge is not true knowledge at all, or is at most only one kind of knowledge. There is a considerable amount of knowledge in our bodies as opposed to minds. The elevation of mind and logic over body and feeling is a conspiracy of masculinity. Haraway's Cyborg Manifesto [1991] is perhaps the best-known version of this, a polemic like Barlow's, but much longer and dressed up in academic observations and questions, but a polemic nonetheless.

How do we respond to such claims? How might we get behind the polemic, and engage critically with them? Is it possible that both claims could contain useful insight?

The opposing of mind and body is seen by Dooyeweerd as inspired by the Matter-Form ground-motive, which is not a 'truth' but a pre-theoretical presupposition. So we may question it, and whether either position has sound foundations for their polar opposition, and we may question some of their assumptions.

Dooyeweerd's theory of inter-aspect dependence, in both directions, makes the pure forms of both untenable. In the foundational direction, information and knowledge (formative and lingual aspects) depend on the physical and biotic aspect and thus on some bodily entity. In the anticipatory direction, the aspects which the feminist thinkers re-emphasise, the biotic and sensitive, only gain their full meaning by anticipating later aspects, which includes the aspects of mind (analytic and formative). This may be what lies behind Haraway's proposal.

Nevertheless it may be that both contain useful insight. We discuss feminist thought in chapter 8. Barlow claims that the 'new reality of mind' implies and demands new understanding of what is legal and moral, and also new legal structures and means of enforcement, which will be worked out by the denizens of Cyberspace themselves without any help from conventional legality. To Dooyeweerd, humanity is mandated to open up the aspects (see Dooyeweerd's theory of progress), and at any point in history we cannot guess what to-be-disclosed meaning an aspect might have in store for us. So our current legal systems and laws and ethical stances should not be assumed to reflect fully the inner normativity of the juridical and ethical aspects. The way it has developed thus far has been strongly influenced by the three dualistic ground-motives, by Aristotle (FMGM), natural law (NGGM) and social contract (NFGM). This takes us into the territory of inscription discussed in chapter 8.

### 5.5.9 On Comparing Computer to the Human Self

Comparisons between computers and human beings can take many forms. One problem we face is that I experiences my own self, including my being-connscious, my knowing, my understanding, my feeling, my thinking, and the like, but I have no access to other selves. I might suppose that other human beings are selves like me, but I cannot assume this about a machine, however cleverly programmed. There have been attempts to circumvent this by closely defining what these might be and then trying to devise tests to determine whether computers can exhibit these. As yet no definition has been put forward that is reasonably universally accepted. The Turing Test, for example, defines 'intelligence' in terms of behaviour, but it defines this only as surface behaviour, and even this is restricted to lingual ability to communicate.

Dooyeweerd believed the human self to be beyond the grasp of theoretical thought, because it is trans-aspectual, and even beyond the grasp of intuitive understanding, because it is supra-temporal. Therefore, while we might be able to define the computer theoretically, we can never furnish a similar theoretical definition of the human self. On the other hand, while we can each experience our own selves, we can never experience what it is like, existentially, to be a computer. Therefore we have no basis for comparing the human self with the computer. Thus any AI challenge that demands comparison between them is meaningless.

This is not, however, dodging the AI challenge. It can be re-interpreted. Instead of trying to find a common theoretical definition of both human and computer, we can compare how human being and computer respond to the aspectual framework of law-and-meaning, and this provides new insight into the challenge.

### 5.5.10 Ideology in AI

Colburn [2000,p.80-81] sums up the debate about whether computers can understand with:

> "If the idea that mental processing can be explained by computational models of symbol manipulation is an AI ideology, there seems to be a countervailing ideology, embodied by Searle and many others, that no matter what the behavior of a purely symbol manipulating system, it can never be said to understand, to have intentionality."

Dooyeweerd sought a philosophical method of dialogue that avoided clashes of ideology, not by denying ideologies but by understanding them using immanent critique and setting them within the same framework, so that we no longer, in Colburn's words "talk past each other".

The basis of ideology is that humankind is inescapably religious (§2.4.1). Dooyeweerd's notion of ground-motives as spiritual driving forces (§2.3.1) can throw light on the diversity of ways in which the artificial intelligence question of whether computers are like human beings is addressed; see Table 5.5.4. Humans exhibit behaviour or property X and computers exhibit Y, and then the question is to what extent and in what ways X = Y. Under the Form-Matter motive, X is mind or information, and Y is physical matter of which the computer is made. Under this dualistic ground-motive, the only way to harmonise X and Y is by giving absolute priority to one, and if necessary reduce the other to it. Materialists give priority to matter while holders of the Cyberspace perspective give priority to mind. Under the Nature-Grace motive, X is sacred 'divine spark' and Y is profanity. The sacred-profane divide implies a normative and not just ontic divide, so those operating under this motive hold as a dogma that they must not attempt to see computers as similar to humans. Under the Nature-Freedom motive, X is non-determinacy and Y is determinacy. Various ways have been attempted to harmonise these. Some merely hold as a dogma that all freedom is illusory. others suggest that even physical behaviour is non-determined, and yet others resort to philosophical idealism. (The unsatisfactoriness of Newell's account of non-determinacy might reflect an attempt to "think together" the two poles of the NFGM, which Dooyeweerd pointed out is always doomed to failure [Dooyeweerd, 1984,I,p.65].)

Searle's answer, that X is biological causality and Y is physical, which are fundamentally different, however, must be understood in a different way, in terms of aspects rather than ground-motives. As we have seen above, he seems to offer no grounds for this difference, holding it as a dogma, and offers no explanation of why it is that biological causality can "process information" while physical causality can only "manipulate symbols". However, Dooyeweerd's notion of aspects solves both these problems. Each aspect enables a different type of 'causality' (repercussion), which accounts for the fundamental difference between biotic and physical causalities. Though Searle holds the dogma that human and computer are fundamentally different, in Dooyeweerd, as we have seen, while the difference is maintained and accounted for, in terms of subject-functioning (Searle was perhaps reaching for what Dooyeweerd offers in his notion of irreducible law spheres). The similarity is also maintained and accounted for, in terms of meaningful-functioning (see Table 5.5.1). And, under Dooyeweerd, Searle is simply wrong to hold that computers can, of themselves, 'manipulate symbols' while holding they cannot 'process information' because both these are object-functionings.

Table 5.5.4 Accounting for Extant Views of Human and Computer

| View | Human | Computer | To harmonise |
|---|---|---|---|
| Form-Matter GM | Mind | Matter | Materialist: reduce mind to matter<br>Cyberspace: deny matter |
| Nature-Grace GM | Sacred | Profane | Dogma: Must not |
| Nature-Freedom GM | Non-determined | Determined | Systems: mystical emergence<br>Quantum: physics not determined<br>Newell: KL incomplete, log.cl. |
| Searle | Biological causality | Physical causality | Dogma: Cannot |
| Dooyeweerd | Subject-functioning | Meaningful-functioning | Cosmonomic notion of law-subject-object relshp |

We can go deeper into the religious root. Most who have addressed the AI question may be seen to have presupposed that we can answer it by seeking some substance, process or type of causality that, in itself and on its own, can explain the difference or similarity -- that is, in terms of Immanence Philosophy. We can see this in Boden's [1990,p.103] suggestion that the main question we must all address is "What things does a machine (whether biological or not) need to be able to do in order to be able to understand?" The strong AI position suggests intentionality may be rooted in physical or logical causal processes, Searle claims it must be rooted in biological causal processes, and Boden herself suggests it is rooted in symbolic causal processes in which "the brain is the medium in which the symbols are floating and in which they trigger each other." [Boden,p.99].

Dooyeweerd rejected any such substance-concept and held that:

> "The inner restlessness of meaning, as the mode of being of created reality, reveals itself in the whole temporal world. To seek a fixed point in the latter is to seek it in a 'fata morgana', a mirage, ... There is indeed nothing in temporal reality in which our heart can rest, because this reality does not rest in itself." [Dooyeweerd, 1984,III,p.109]

Both supporters and opponents of strong AI seek a "fixed point" in some kind of "thing-reality" (including causality etc.), but because this is a mirage, both have problems in accounting for meaning (semantics) and must rely on dogma or mystical connection. But Dooyeweerd's approach, which presupposes meaning, does not need to find such a connection, since they occur especially in the inter-aspect relationships.

## 5.6 CONCLUSION

One might expect that trying to understand the nature of computers, information and programs would be a theoretical exercise, especially if three decades of debate in artificial intelligence are taken as a model. So what is everyday experience in this area of research and practice? Is everyday experience even possible?

The starting point in trying to understand the nature of computers is to take them as they present themselves to us in our everyday lives as users, developers, etc., and to seek an understanding of them that pertains regardless of application.

But 'as computers present themselves to us' is so closely tied up with their application, that we have to be careful in how we approach the issue of their nature. So we had first to settle the issue of what we mean by the nature of a thing. After noting a number of problems with conventional assumptions about Being as such, we turned to Dooyeweerd's approach, which founds Being in cosmic meaning. The first principle of the framework developed here arose from Dooyeweerd's contention that in everyday experience things are not experienced as completely separate entities:

# The nature of computers is to be understood by reference to human beings. It functions as object, not subject, in all but the physical aspect.

Computers exist qua computer by virtue of human subject-functioning in various aspects.

This principle, seeing the computer as functioning as aspectual object, proved useful, later in the chapter, in throwing fresh light on the artificial intelligence question of whether computers can understand, by reference to Searle's thought-experiment, the Chinese Room. That Dooyeweerd's non-Cartesian subject-object relation is grounded in cosmic law and meaning rather than in self-dependent entities, as both Cartesian and anti-Cartesian notions are, allows us to meaningfully state that computers can understand, as long as this is recognised to be object-functioning rather than subject-functioning. Moreover, Dooyeweerd's delineation of the ground-motives could account for three dualistic approaches to the AI question. By these means, the AI question can be approached in a completely new way, with a hope of resolution.

Referring back to chapter 4, it was determined that the subject-functioning that is relevant here was HCI, which is qualified by the lingual aspect regardless of application. This leads to the second, and central principle:

# The being of a computer is multi-aspectual, multi-levelled; a computer is a meaningful whole constituted of a number of aspectual beings, in which the lingual aspect is key.

It was noted later that this accords very closely to Newell's [1982] notion of computer system levels, which likewise arose from his reflection on the everyday life of AI researchers and practitioners. Newell's proposal was discussed and enriched.

This principle also offers a very practical guide to teaching many computer-related topics: separate the issues out into each aspect.

That the computer is a multi-aspectual being immediately raises the question of the relationship between the beings, and two were found:

# While a part-whole relation may be found among beings within each aspect, the aspectual beings of different aspects are bound together in the whole by foundational enkapsis, in which inter-aspect dependency plays an important part.

This enabled us to address the nature of implementation of the various levels (aspects) of the computer, each in terms of earlier aspects. On this basis, a number of issues could be considered, including the freedom that implementors have of each aspect, the possibility of virtual data, the impossibility of interpreting the computer seen from one aspect mechanically from a description at another, randomizing and file compression, and an understanding of errors.

Finally, the nature of information and program were considered. Information was likewise understood in multi-aspectual terms, while programs could be seen either as a virtual law side, and also in terms of Dooyeweerd's discussion of the nature of performance art.

Some benefits of this framework are that it can throw fresh light on issues where debate has deteriorated, or is likely to deteriorate, into dogmatic positions, so that the opposing positions may begin to understand and accept something of the validity of each other without negating their own positions. It is a fuller framework, which expresses something of the richness of our everyday experience of computers.

With this understanding of the nature of computers, information and program, combined with an understanding of usage (chapter 4), it is now possible to make the consideration of IS development (chapter 6), the creation of basic technologies (chapter 7) and societal views of ICT (chapter 8), somewhat richer than it might otherwise have been.

## References

Alavi, M., & Leidner, D. E. (2001). Review: knowledge management and knowledge management systems: Conceptual foundations and research issues. MIS Quarterly, 25(1), 107-136.

Barlow, J. P. (1996). A declaration of the independence of cyberspace. Retrieved March 16, 2005, from http://homes.eff.org/~barlow/Declaration-Final.html

Basden, A,, & Burke, M. (2004). Towards a philosophical understanding of documentation: a Dooyeweerdian framework. Journal of Documentation, 60(4), 352-370.

Basden, A., & Brown, A. J. (1996). Istar - a tool for creative design of knowledge bases. Expert Systems, 13(4), 259-276.
Boden, M. A. (1990). Escaping from the Chinese room. In M. A. Boden (Ed.), The philosophy of artificial intelligence (pp. 89-104). Oxford, England: Oxford University Press.
Colburn, T. R. (2000). Philosophy and computer science. Armonk, NY: M. E. Sharpe.
De Raadt, J. D. R. (1991). Information and managerial wisdom. Pocatello, ID: Paradigm Publications.
Dennett, D. C. (1978). Brainstorms: Philosophical essays on mind and psychology. Montgomery, VT: Bradford Books.
Dennett, D. C. (1998). The intentional stance. Cambridge, MA: Bradford Books/MIT Press.
Dollar, C. M. (2000). Authentic electronic records: strategies for long-term access. Chicago: Cohasset.
Dooyeweerd, H. (1984). A new critique of theoretical thought (Vols. 1-4). Jordan Station, Ontario, Canada: Paideia Press. (Original work published 1953-1958)
Haraway, D. (1991). Simians, cyborgs and women. New York: Routledge.
Jennings, N. R. (2000). On agent-based software engineering. Artificial Intelligence, 117, 277-296.
Langefors, B. (1966). Theoretical analysis of information systems. Lund, Sweden: Studentlitteratur.
Milewski, A. E. (1997). Delegating to software agents. International Journal of Human-Computer Studies, 46, 485-500.
Newell, A. (1982). The knowledge level. Artificial Intelligence, 18, 87-127.
Newell, A. (1993). Reflections on the knowledge level. Artificial Intelligence, 59, 31-38.
Schuurman, E. (1980). Technology and the future: A philosophical challenge. Toronto, Ontario, Canada: Wedge.
Searle, J. R. (1990). Minds, brains and programs. In M. A. Boden (Ed.), The philosophy of artificial intelligence (pp. 67-88). Oxford, England: Oxford University Press. (Original work published 1980 in Behavioral and Brain Sciences, 3, 417-424)
Stockton, F.R. (1895). A Chosen Few: Short Stories. New York: Charles Scribner’s Sons.
Tuomi, I. (1999). Data is more than knowledge: Implications of the reversed hierarchy for knowledge management and organizational memory. In Hawaii International Conference on Systems Sciences, Proceedings from the Thirty-Second HICSS (pp. 147-155). Los Alamitos, CA: IEEE Computer Society Press.
Wilber, K. (2000). A theory of everything: An integral vision for business, politics, Science and Spirituality. Boston, MA: Shambhala.
Winograd, T., & Flores, F. (1986). Understanding computers and cognition: A new foundation for design. Reading, MA: Addison-Wesley