

# Some Weird Stuff

---

## Richard Feynman

For any person wishing to carry a mapper's strengths into the workplace, the life and work of the physicist Richard Feynman is worth studying. He told stories. The Spencers' Warbler was a bird identified for him by his father. The name was made up. His father then made up names for the bird in many other languages, and pointed out that young Feynman knew no more than when he started. Rote-learning names of things means nothing. Only looking at what the bird itself is doing tells one anything about it.

He was utterly honest and saw through artificial complexity by always insisting on simplicity and facts. See his personal version of The Challenger Report, contained in his book *What Do You Care What Other People Think?*.

He used simple, humorous, curious language, filled with little pictures and enthusiasm. His techniques for puncturing pomposity were unrestrained.

His *Lectures on Computation* have recently been published, and are worth reading, as is everything he ever published, from *Six Easy Pieces*, to the *Red Book Lectures*. James Gleik's *Genius* and the Gribben's *Richard Feynman* are rewarding biographies.

Get hold of his stuff and read it.

## George Spencer-Brown

*The Laws of Form*, by George Spencer-Brown is a little book of mathematics and commentary that is described by modern logicians as containing a form of `modal logic', characterised by having the rules of the logical system applying differently in different places, in a manner defined by the rules of the logic itself.

From the point of view of a programmer, there are two aspects to this book that will certainly stimulate thought. In the main text, the author shows how to do predicate logic with just one symbol, offering a deeper view of `fundamental' logical and computational operations such as NOT, OR AND, XOR than one might have guessed existed.

Then there are the notes, simple and profound thoughts that one returns to again and again, often informed by the technique of doing predicate logic with one symbol, that can be thought of as simply cutting a single plane into two pieces, so that there are two distinguished things, and thus something to talk about. For example, the author says,

In all mathematics it becomes apparent, at some stage, that we have for some time been following a rule without being consciously aware of the fact. This might be described as the use of a covert convention. A recognisable aspect of the advancement of mathematics consists of the advancement of the consciousness of what we are doing, whereby the covert becomes overt. Mathematics is in this respect psychedelic.

Or try,

In discovering a proof, we must do something more subtle than search. We must come to see the relevance, in respect of whatever statement it is we wish to justify, of some fact in full view, and of which, therefore, we are already constantly aware. Whereas we may know how to undertake a search for something we can not see, the subtlety of the technique of trying to `find' something which we already can see may more easily escape our efforts.

Or,

Discoveries of any great moment in mathematics and other disciplines, once they are discovered, are seen to be extremely simple and obvious, and make everybody, including their discoverer, appear foolish for not having discovered them before. It is all too often forgotten that the ancient symbol for the pre-nascence of the world is a fool, and that foolishness, being a divine state, is not a condition to be either proud or ashamed of.

Unfortunately we find systems of education today which have departed so far from the plain truth, that they now teach us to be proud of what we know and ashamed of ignorance. This is doubly corrupt. It is corrupt not only because pride is in itself a mortal sin, but also to teach pride in knowledge is to put up an effective barrier against any advance upon what is already known, since it makes one ashamed to look beyond the bonds imposed by one's ignorance.

To any person prepared to enter with respect into the realm of his great and universal ignorance, the secrets of being will eventually unfold, and they will do so in a measure according to his freedom from natural and indoctrinated shame in his respect of their revelation.

In the face of the strong, and indeed violent, social pressures against it, few people have been prepared to take this simple and satisfying course towards sanity. And in a society where a prominent psychiatrist can advertise that, given the chance, he would have treated Newton to electric shock therapy, who can blame any person for being afraid to do so?

To arrive at the simplest truth, as Newton knew and practiced, requires years of contemplation. Not activity. Not reasoning. Not calculating. Not busy behaviour of any kind. Not reading. Not talking. Not making an effort. Not thinking. Simply bearing in mind what it is one needs to know. And yet those with the courage to tread this path to real discovery are not only offered practically no guidance on how to do so, they are actively discouraged and have to set about it in secret, pretending meanwhile to be diligently engaged in the frantic diversions and to conform with the deadening personal opinions which are being continually thrust upon them.

As a beautiful summary of the mapper/packer communication barrier that we have discussed at such length, one can hardly do better than that! Finally, there is a vision of the power of the mapping cognitive strategy, as it continues to seek for ever deeper structure behind the phenomena it regards, offered by way of what we get by making a single distinction in the void,

We are, and have been all along, deliberating the form of a single construction ... notably the first distinction. The whole account of our deliberations is an account of how it may appear, in the light of various state of mind which we put upon ourselves.

Elsewhere he says,

Thus we cannot escape the fact that the world we know is constructed in order (and thus in such a way as to be able) to see itself.

Richness from ultimate simplicity. The limit of complexity cancellation, and the art of using the triangle of creativity to place the Knight's Fork of our perception at the correct level of abstraction for our purposes. As programmers, we work in, and by our every deed prove the unification of, exactly the same creative space as the most abstracted of mathematicians and lyrical of poets. Remembering George Spencer-Brown, look at this poem by Laurie Lee, and ask if your code has ever drawn structure from domain, done all that has to be done, and outroed so perfectly?

*Fish and Water*

A golden fish like a pint of wine  
Rolls the sea undergreen,  
Glassily balanced on the tide  
Only the skin between.

Fish and water lean together,  
Separate and one,

Till a fatal flash of the instant sun  
Lazily corkscrews down.

Did fish and water drink each other?  
The reed leans there alone;  
As we, who once drank each other's breath,  
Have emptied the air, and gone.

## Physics Textbook as Cultural Construct

We are regularly invited to see the world in a certain way, by users who believe they understand their world, by style and approach gurus, by our own preconceptions. We are continually challenged to see the world as it is, such that we make its representations in our systems as simple as possible. Just as one has to see the quality plateau (albeit only once) before one can recognise it, so one has to 'Walk around the side of the Gone With the Windbreak and see how many times they lit the fire'; one has to see a supposed solid reality questioned, before one can know what this is about.

There can't be much more solid than A Level Physics: anyone who says that that is a cultural construct, a social agreement between cynical physicists to make the world obscure to civilised people with media degrees would clearly have to be off their rockers. The strange thing is, some people genuinely do argue that the laws of physics are made up by physicists rather than discovered, and they should be constrained to make them up differently!

The real tragedy for these prattling fools is that if only they were to study some physics, they might have discovered that although the laws of physics were in place long before the physicists that study them and are quite independent of the opinions of the physicists, the perception of the universe that we draw from these laws may well be a cultural construct.

To explain this amazing claim, we need to refer to three physicists. Isaac Newton discovered modern mechanics, and actually recorded his discoveries mainly in Latin prose, not in the symological style we use today. That was invented by the Victorian Oliver Heavyside, and what we usually refer to as 'Newtonian' physics is nearly always in fact the Heavyside rendition of Newton's physics. Richard Feynman was a physicist of modern times, who attempted to summarise what was known as elegantly as he could for undergraduates in the *Red Books*. Where things get interesting is when we compare the ordering of the tables of contents in the *Principia* of the genius Newton, the parts of the genius Feynman's *Red Books* that were known to Newton, and the parts of *Advanced Level Physics* by Nelkon and Parker (the standard British textbook), again that were known to Newton.

### Principia

- Newton's Three Laws of Motion
- Orbits in gravitation (with raising and lowering things)
- Motion in resistive media
- Hydrostatics
- Pendulums
- Motions through fluids.

### Red Books

- Energy
- Time and distance
- Gravitation
- Motion
- Newton's Three Laws

- Raising and lowering things
- Pendulums
- Hydrostatics and flow.

## Advanced Level Physics

- Newton's Three Laws
- Pendulums
- Hydrostatics
- Gravitation
- Energy

What seems to be distinctive about *Advanced Level Physics* is that its mechanics builds up the complexity of the equations of Heavyside's system, whereas the two other works are motivated by different intents.

Newton starts with his Three Laws, while Feynman gets energy into the picture really early and leaves the Three Laws until later. But once they have defined some terms to work with, both geniuses start by telling us of a universe where everything is always in motion about everything else, and then fill in that picture. They do this long before they discuss pendulums, which are arithmetically much easier, but are a special case compared to the unfettered planets in their orbits.

*Advanced Level Physics* puts pendulums before gravitation, indeed deals with the hydrostatic stuff both geniuses leave until very late, before it even mentions gravitation, by which time, we suggest, the student has learned to perform calculations in exams as efficiently as possible, but has possibly built a mental model of a universe of largely static reference frames with oddities moving relative to them.

Algebraically inconvenient though it may be (and while Newton's prose might not be influenced by algebraic rendition, Feynman obviously had to consider it), both geniuses want to get the idea that everything moves, in right at the start.

Might it be possible to learn even physics the wrong way, and end up able to do sums concerning the goings on within the universe, but still with a warped and confused view of it?

## Are Electrons Conscious?

In *The Quantum Self*, Danah Zohar considers some questions relating to the nature of consciousness. One idea from consciousness studies suggests that the phenomenon of consciousness emerges from complex relationships between things that are not, in themselves, conscious. This begs the question of how little consciousness one can have. Can an electron, jiggling about and doing its mysterious, wavelike thing, be a little bit conscious?

We have raised Zohar's question not to attempt to answer it directly, but to try to approach it from another direction. And as with all this 'Weird Stuff', the intent is not to provide information, but to demonstrate just how close the day to day work of a programmer really is to the highest arts and the deepest mysteries.

We will start by doing you the courtesy of assuming you are conscious. Imagine you make a study of synchronised processes sharing resources. As a good mapper, you research the literature, and contemplate what others have said. You also try some experiments yourself. Pretty soon you start to see the deep invariant patterns, both successful ones and unsuccessful ones. You come to realise that a potential deadlock situation is a potential deadlock no matter how it is decorated with complexity. You also come to recognise a potential livelock when you see one.

For those readers that have not made this study, please note that you should, as too many programmer hours are wasted on this stuff, but here's a summary of deadlock and livelock. A deadlock arises when two (or more) processes end up halted, mutually waiting on each other. For example, one process might acquire exclusive access to the customer database, while another acquires exclusive access to the stock database. Then each process attempts to get exclusive access to the database it hasn't got. Neither processes' request can be fulfilled, because the other process already has the exclusive access

requested. So the database manager just leaves both calls pending, both processes asleep, until the requests can be fulfilled. Of course, this will never happen, because neither sleeping process can relinquish the database it already has, so both sleep forever. The easiest way to avoid this situation on a real project incidentally, is not particularly clever. The word customer sorts before the word stock, so make it a mass drinks buying offense to ever acquire the stock database before the customer database, even if this means that situations emerge where one only has access to the stock database already, and so one has to relinquish stock, acquire customer, acquire stock. It's worth it and let's face it, either access will be granted instantly or some other necessary process will get in there and the cycles will be used well.

A livelock is a kind of variation of a deadlock where (for example) each process returns with a failure code instead of sleeping, and tries to help by relinquishing the resources it has got and then carrying on with its shopping list. So both processes chase each other's tails until one or the other manages to get enough cycles in one go to acquire both resources at once, and break the cycle.

So now you know livelocks. From bitter experience you know livelocks, and you recognise a potential livelock when you see one. Now imagine you are planning to meet a friend. You aren't sure which of two bars you will want to meet in, because one or the other is always lively when the other is like a morgue, and you can never tell which way around it will be. You don't know which of you will arrive first. The two bars are on opposite sides of the same city block. Of course, you know livelocks. As a furry animal running around planet Earth you aren't going to have your say, opportunities to mate, reduced by a stupid livelock wherein you both chase around in circles between the two bars looking for each other. When you make the date, you are the person who says, 'And if you want to check the other bar, walk around the river side of the block so I'll see you if I pull the same trick!'

That's you. It's the kind of person you are. The person you are going to meet has already been attracted by this simultaneously imaginative and sensible aspect of your character, and approves the plan.

So what we understand and what we are are intertwined. When you understand livelock, understanding of livelock becomes a part of your consciousness - the awareness that this universe does that kind of stuff, so you deal with it.

Now imagine that you are asked to look at the information flows around a major corporation, and propose a network management algorithm that optimises corporate bandwidth. You perform a mapper study, as you did with livelock, and eventually you experience insights (problem quakes) that allow you to see an elegant, robust and extensible network management strategy.

Now this strategy, just like livelock, is a part of you. When you see bits of the problem repeated elsewhere, bits of your strategy will be obviously applicable, although at the time, you may swear blind that 'It's just so!', and be unable to say why. So when you subsequently capture your elegant, succinct understanding in a programming language and set it running, to what extent is there a copy of a little bit of you running the corporate comms, 24 hours a day?

This is a deep question, and not at all easy to understand. To see it explored somewhat, look at Marvin Minsky and Harry Harrison's science fiction novel, *The Turing Option*.

For the traditionally philosophically minded, we might make an additional observation in this regard. Usually the essential, such as the Platonic abstraction of 'two-ness' is never seen directly, but only through the phenomenal, such as two dogs, two legs or eyes. It is usually considered that the essential in some way proceeds the phenomenal, because the abstraction of two-ness remains even when there isn't a pair of anything in view. The phenomenal is usually, if covertly (in Spencer-Brown's use of the word) seen as proceeding from the essential.

Now consider what happens in the writing of a one-bit program. The triangle of creativity, comprising problem dynamics, system semantics and desire, is certainly phenomenal, because it takes place in the head of the programmer, who has to be actually and physically in existence. However, the triangle of creativity leaves as its product the Knight's Fork, which is an essential mapping of problem dynamics to system semantics. The Knight's Fork, which is essential, in this case is in the image of, and proceeds from, the triangle of creativity, which is phenomenal. Could this reversal of the usually accepted direction of ontological priority be connected with the strange way that a ROM chip gets a peculiar kind of negative entropy added to it as it passes through our hands?

# Teilhard de Chardin and Vernor Vinge

Pierre Teilhard de Chardin was a palaeontologist and Jesuit who wrote *The Phenomenon of Man* in the mid-1950's. By deducing a pattern from fossil evidence and filling in the black-box properties of the parts of his model that he didn't understand with semi-allegorically, semi-religiously worded speculations, he arrived at an unusual view of evolution that proposed a predictable direction of its future course. Although Teilhard de Chardin's thought was very peculiar at its time, his ideas have been sliding towards the centre of some peoples' view of what is happening with technology at the moment and the universe in general. The work hasn't changed, it's just that we are picking up evidence suggesting that the mental model of evolution that it proposes happens to be close to the truth.

Teilhard de Chardin identifies a raising in complexity of forms, first with the aggregation of atomic matter in the formation of planets (geosphere), then upon the geosphere the appearance of life (biosphere), then the development by life of consciousness. He suggests that the next stage is the interaction of conscious units to create a 'noosphere', which will be a whole new ballgame using the underlying minds as a platform, as the minds use the brains and the brains use the molecules. The behaviours and relevant environmental influences of minds, brains and molecules are totally different, and we can expect the next stage to be no different.

He suggests that there does not have to be any coercion involved in the necessary adoption of co-ordinated states by enough individual minds for an aggregate identity to form - perhaps this is what we see in a 'gelled team', which shares a mental model about what the hell is going on. He proposes that the ultimate confluence will be what he calls the Omega Point, where co-ordinated interaction of the constituent minds of the noosphere overwhelms non-coordinated action and a new state emerges.

He was not without his critics - Sir Peter Medawar wrote a scathing attack that focussed on the language changes at the interfaces between the solid evidential parts of the argument and the processes of unknown mechanism fitted in between them. In particular Medawar became very excited about Teilhard de Chardin's use of the word 'vibration' where it was clear that the words 'coupling' or 'constraint' could have been used, and might not have excited Medawar quite so much. The trouble is, mappers have to work with things they don't understand, so the language inevitably gets a little fluffy in places. That's where new theories come from (and one might say that a program is the programmer's theory of the problem domain). Unfortunately this kind of language drives some people crazy, even though most of the good stuff has some of it kicking around, if only in the form of saying that things 'want' to do this or that, and filling in the unknown mechanism with an anthropomorphism that is just as silly, applied to an electron, let alone an ant, as proposing an 'ineffable spirit', but is for some reason more acceptable.

For an extreme example, listen to Newton, slagging off the bits he could see were missing from his own physical picture, but could not explain the mechanism of (which was the whole point of course)...

Of course, it's a well known fact that Newton spent much of his life 'messing around with theology'!

Vernor Vinge is an Associate Professor of Mathematical Sciences at San Diego State University, and one of the best science fiction writers around. In his famous 'Singularity Paper', (use the WWW and the SF books *Across Realtime* and *A Fire Upon the Deep*, he proposes that the intelligence of beings on this planet will increase, either by improving human brains genetically, or by giving them hardware enhancements, or by building new trans-human computer architectures. After this, networking and a new agenda that comes from seeing more will create a world that we are inherently incapable of imagining in our current state.

There is a striking similarity between the ideas of Teilhard de Chardin and Vinge, only by moving evolution into the fast-burn of software, we shrink the millions of years of organic evolution required by Teilhard de Chardin for the construction of the noosphere to the thirty proposed by Vinge.

But don't take our word for any of this stuff - check it out, see if it gives you a new perspective on what the universe is doing when you are programming, and above all, think about it if only for practice!

# Society of Mind

Marvin Minsky proposed in *The Society of Mind* that the phenomenon of human consciousness emerges from the interaction of numbers of unconsciousness processing agents that run like co-processes in the brain, each with its own triggers and agendas. The agents are then connected up and arbitrated via a 'netiquette' that allows them to determine the course of action the organism as a whole will take. When we feel ourselves exercising free will in pursuing our whims, we are in fact simply enacting a decision that has already been arrived at by the collective of agents. The model certainly has its attractions, and gives a basis for the drives that we use our creativity and intelligence to accomplish, but doesn't seem to give a useful description of the creativity and intelligence themselves. With these generalised cognitive faculties, the brain seems to be used as a directable general purpose pattern recognition device whose internal representations are coupled to the sensory components indirectly, at least such that the abstract and the concrete can be considered in the same terms.

The relationship between the society of mind model of cognition and motivation and the general purpose faculties mirrors the relationship between what we have called the packing and mapping strategies, and there is a further parallel with two simple approaches to managing data in computer system design.

Hash buckets operate by abstracting some sort of a key out of the data - perhaps by taking a 20 character name field and adding up the numeric value of all the characters. That number can then be used to index into a table and find the full record. Real hashing algorithms are designed to maximise the spread in the resulting number from typical input data, and must cope with situations where the hash bucket is already full by putting references to several records in them, such that retrieval involves then checking the full key on each record in the bucket. Hash buckets are often very effective in simple situations, and are reminiscent of packing, where some abstraction of the situations encountered is used to trigger 'appropriate action'. In packing, hash collisions seem to be poorly handled. They will not even be noticed unless one or more participants will suffer short term loss due to 'appropriate action'. Then an 'argument' will ensue, where one packer points to one way of abstracting the hash key from the situation and argue that it is 'the case', while another will point to another hashing algorithm and argue that no, their way is 'the case'. This is not productive and shows a breakdown of the strategy above a certain level of problem complexity, where we are just trying to cram too much variation into too few hash buckets and have not developed the skills to do the significant amounts of full key examination that is then necessary.

Object models allow the data structures held in the computer to grow in complex and dynamic ways, constrained to the semantics of the modelled objects. The shape of the whole data structure can change completely during processing, and retrieval always stays 'natural' in that the data are where they 'ought' to be - they are all directly associated with an appropriate other datum. Hence there is no complexity introduced by a foreign algorithm such as hashing to be cancelled by something else such as exhaustive key comparison. Above a certain level of complexity, object models are more suitable than hash buckets, but there is no doubt that they are actually harder to implement. The reason why we can use them at low cost today is that we get a lot of specialist support from our languages for describing objects, and our operating systems for free memory management. Object models seem so similar to the mapper strategy that we have described mapping as the attempt to construct a viable object model of the problem domain.

These parallels between functional (society of mind and pattern recognition), subjective descriptive (packing and mapping) and computational (hashing and object modelling) models of consciousness suggest that there may even be a neurological correlate to the mapping and packing strategies we have described. We certainly know that early stimulation of infants causes increased neuron growth and interconnection in infant brains, and this correlates with higher 'intelligence' (whatever that may be) in adult life. Whatever 'intelligence' is, the kind of cognitive and problem solving skills that are tested have little place in the Taylorist, packer workplace, where the whole idea is to deskill and constrain behaviour.

Perhaps the question at the start of the Information Age is: 'What part of your brain is it appropriate to use at work?'

## Mapping and Mysticism

Right at the beginning, we looked at two different ways of going about solving problems. Packing was characterised as a socially conditioned habit of accreting 'knowledge packets' that specify 'appropriate action', and not examining or reconfiguring the relationships between the knowledge packets. The strategy degenerates into kludging reality to fit the known packets and blaming luck when things go wrong. Mapping on the other hand, involves putting investment into

building an internal object model of the world as it is perceived and getting leverage by identifying deep structure. Mapping can be developed by learning techniques that help the exploration of conceptual spaces and help one recognise what it is that one is actually seeing happening in front of one's own eyes, by recognising the deep structure patterns in the goings on. Mappers can respond flexibly and are the only people in a position to propose new approaches. They can learn vastly more quickly than packers, and unless they are seeing deep structure, they are seeing as yet unsolved mysteries. The experience of mappers and packers may be quite different, in exactly the same circumstances.

Mapping is the natural state of people, and everyone is a mapper at heart. Unfortunately, societies the world over developed an alternative, which we have called packing, possibly at around the same time as we discovered agriculture approximately 6,000 years ago. It could not have been before that - a pre-agrarian packer confronting a wild animal on the hunt could hardly have fared well by sticking his nose in the air and claiming that the animal was failing to follow procedure!

The alternative involves convincing people that good living consists of following prescribed procedures, and suppressing any alternatives. It must have brought benefits to new societies based around the raising of crops, where significant tedious work must be done in the fields, and if things are tight, the only thing one can do is plod, plod, plod, until harvest when more crops will be available. Packing thus involves socialising the young into the packer mindset, and constructing a society where reality consists of the packer approach and a set of knowledge packets, and nothing else. Any person who suspects that there may be other ways of looking at things is then at odds with every member of the society he or she finds themselves in, at odds with the inefficiencies of packer society and the ritualised manner that even social occasions are brought down to. The dissenter may be ascribed weird properties such as magical powers if they are lucky enough to implement some common-sense ideas, or madness if the surrounding packers manage to sabotage their deviation in time. Most people would not even believe that any way of approaching the world other than packing could even exist.

Today, there is little call for stoop labour in the developed world, but a significant need for fully aware people to create the new programs that will run our automation. Only natural mappers have the pattern recognition skills essential for writing computer programs.

For much of its existence, the packer strategy has probably served its users pretty well, keeping order in the fields and early factories, and ensuring that the simple manual labour that was essential to survival was performed. In the subsistence conditions that prevailed, perhaps the literary arts could have been better served, but since the invention of the printing press, there have actually been more poets than printing presses, so perhaps there has not even been a great cost there. But by the start of the 20th century, the age of industrialisation had made packing a dangerously inefficient strategy. We were just too wealthy. Our engines could allow us to do things undreamed of by previous generations, and we needed understanding to guide our use of them. Trapped in a packer mindset, and in possession of knowledge packets inappropriate to industrialised societies, Europe was torn apart as millions went to war with internal combustion engines, tracked vehicles, barbed wire, machine guns, mustard gas, aeroplanes and other equipment that distorted the pre-industrial knowledge packet that 'War is an extension of diplomacy by other means' out of any reasonable diplomatic definition of objectives, on grounds of cost alone.

Now the cracks are really starting to show. We have achieved the dream of ages, and have abandoned the need to work for millions, freeing up their time to do what they would wish. Yet we see this as unemployment, and furthermore we keep millions working away in what used to be low-overhead jobs, just manipulating the tokens of an agrarian economic system. There are so many of these non-productive jobs that it is actually hard to see it, but every supermarket checkout operator, bank cashier, ticket inspector, financial advisor, tax collector, accountant and on and on is in fact engaged in non-productive labour. Only a tiny fraction of the population are doing any work necessary for the maintenance of our material lifestyles, yet still we believe ourselves to be in scarcity!

Even the currently visible stresses, far worse than any in history (packing always leads to a kind of stupidity when decisions about unusual circumstances have to be made), cannot point the way back to mapping in packer language. One might say that it is a function of packer language, evolved over millenia, to prevent the discussion of mapping! So in times gone by, returning to mapping must have been very rare indeed. If the effectiveness or acceptability of a mapping approach to a problem is a matter of opinion, the opinion of the majority of packers will always be that it doesn't matter that the result was obtained, because it wasn't done 'properly'.

Only today is there a real opportunity for an individual to practice mapping and get the realistic feedback that is essential for learning. That is because only mappers can program computers. If a person, still in the packer mindset, follows the

procedure and translates a requirement, the result is likely to be a mess. At this point, he or she could blame the compiler, the operating system or the user, but possibly might just recognise that the computer really is, with utter faithfulness, reflecting what it has been told. So the individual can accept that it is they, and they alone, that must understand the problem dynamics and system semantics. Here begins many late nights, and the opening of the road to really thinking, rather than performing the abberation that is packing, and which the packer majority calls normal.

From this perspective, it's interesting to look at several strands of previous thought that attempt to describe the experience of mapping in cultures where one cannot just say 'The program works!', and have a very strong argument on one's side, in language that mappers can understand in terms of shared subjective experience of playing with representations of reality in one's head until they are correct enough to be useful, and which packers cannot understand at all. Perhaps it is not surprising that many great programmers have interests in these strands of previous thought..

We have already discussed the nature of alchemy as an internal journey that changes the operator's view of the world - the basic technique of mapping. Alchemical traditions likely spread into Europe from Moorish Spain through people like Roger Bacon.

In *In Search of the Miraculous*, PD Ouspensky records some conversations with GI Gurdjieff, which took place in Russia in 1915. A strange figure who made a remarkable impact, Gurdjieff told that he had spent many years studying mystical traditions, Ouspensky records,

In all there are four states of conciousness possible for man... but ordinary man... lives in the two lowest states of conciousness only. The two higher states of conciousness are inaccessible to him, and although he may have flashes of these states, he is unable to understand them, and he judges them from the point of view of those states in which it is usual for him to be.

The two usual, that is, the lowest, states of conciousness are first, sleep, in other words a passive state in which man spends a third and very often a half of his life. And second, the state in which men spend the other part of their lives, in which they walk the streets, write books, talk on lofty subjects, take part in politics, kill one another, which they regard as active and call 'clear conciousness' or 'the waking state of conciousness'. The term 'clear conciousness' or 'the waking state of conciousness' seems to have been given in jest, especially when you realise what clear conciousness ought in reality to be and what the state in which man lives and acts really is.

The third state of conciousness is self-remembering or self-consciousness or conciousness of one's being. It is usual to consider that we have this state of conciousness or that we can have it if we want it. ur science and philosophy have overlooked the fact that we do not possess this state of conciousness and that we cannot create it in ourselves by desire or decision alone.

The fourth state of conciousness is called the objective state of conciousness. In this state a man can see things as they are. Flashes of this state of conciousness also occur in man. In the religions of all nations there are indications of the possibility of a state of conciousness of this kind which is called 'enlightenment' and various other names but which cannot be described in words. But the only right way to objective conciousness is through the development of self-consciousness. If an ordinary man is artificially brought into a state of objective conciousness and afterwards brought back to his usual state he will remember nothing and he will think that for a time he had lost conciousness. But in the state of self-consciousness a man can have flashes of objective conciousness and remember them.

The fourth state of conciousness in man means an altogether different state of being; it is the result of long and difficult work on oneself.

But the third state of conciousness constitutes the natural right of man as he is, and if man does not possess it, it is only because of the wrong conditions of his life. It can be said without any exaggeration that at the present time the third state of conciousness occurs in man only in the form of very rare flashes and that it can be made more or less permanent in him only by means of special training.

This certainly sounds like packing corresponds to the second state, mapping to the third state, and whatever happens in problem quake to the fourth state. Happily the difficulties Gurdjieff described are greatly mitigated today by kind employers who are willing to pay us high salaries to sit in front of training machines all day. If we can adopt third level

language at work instead of second level, we will be able to repay these kindly people by writing lots of nifty computer programs for them.

We should say in fairness, that while much of *In Search of the Miraculous* is directly accessible in terms of the mapper/packer model, much is not. There is also a system of 'Hydrogens' that seems to be utterly unconnected to particle physics, which supposedly describes the structure of the universe. It does however bring fractal structure and attractors to mind, and purports to be a world-view that enables an individual to enjoy vastly increased options by 'freeing himself from general laws' in a fashion not amenable to reductionist description. We can't make head nor tail of this stuff, but having seen mappers and packers in the work only after finding them amongst the computers, we suspect it may be worth... contemplating.

In Islam, there is the concept of two Korans. There is the written Koran, recorded by the Prophet at the command of God, and the manifest Koran, which is the world about us created by God. It is the duty of every person who enjoys the luxury of improving himself by spending his time studying these works of God, to pass on his findings in a manner accessible to all. Perhaps this beautiful idea, which allows the student to acknowledge his ignorance by setting up a hopeless direct competition with God that everyone is bound to lose anyway, and then teaching that it is the student's spiritual duty to reduce this ignorance, might have something to do with Islam's staggering contributions to our field. We all know where algebra and algorithms came from!

In China there is the ancient Taoist tradition, which also suffers from a communication problem - the *Tao Te Ching* begins,  
The Tao that can be told is not the eternal Tao.

Taoists concentrate in finding the deep structure of the deep structure, and obtaining maximal leverage by 'right action'. A Taoist does not limply 'go with the flow', he has a clear (and hence non-contradictory or perverse) understanding of what he wishes to accomplish, and looks for the right point to apply influence, by looking at the structure of the interconnected phenomena that he is interested in. The right action might then be a swift kick at just the right place! In common with all mystical traditions, Taoists have no time for pomposity whatsoever.

When Taoism met Bhuddism, Zen appeared. From the mapper/packer perspective, Zen might be described as a specialist set of mapper techniques and building blocks, that allow exploration of deep structure that is often counter-intuitive to someone afflicted with the packer mindset. When Zen asks, 'What is the sound of one hand clapping?', it is saying that the clap is to be found in neither the left hand, nor the right, but in the interaction between them. Many great programmers, especially Artificial Intelligence workers, love tickling themselves with Zen koans.

Alchemy, Taoism and Zen are all mystical teachings that have no supernatural component to them at all. They discuss the state of mind of the practitioner, and thus increase the available options by removing the rigid mass of preconceptions that packing produces. As Kate Bush (another favourite amongst programmers) put it,

*Don't fall for a magic world  
We humans got it all  
Every one of us  
Has a heaven inside.*

But despite their practical emphasis, they all have to use allegorical language to discuss the subjective mapper experience. Ancient allegorical language that makes no sense at all to packers is easily mistaken for religion, and in the 19th century other workers attempted to erect strictly secular descriptions of what is going on.

The philosopher Frederic Nietzsche ran up against the mapper/packer communication barrier in a big way, and caused great excitement amongst his local packer community by declaring that the Superman was not bound by mere laws. He died in a mental asylum, but not before making a significant impact upon philosophy.

Nietzsche was concerned with the difference between a person who has reached his own potential and someone who lives in a socialised packer reality. He really didn't like the snivelling, envious, spiteful, small minded common man that he held up against his Superman at all. He has been getting renewed interest from people involved in TQM recently.

Sigmund Freud interviewed large numbers of Viennese middle class women and came up with an original psychoanalysis that included an idiosyncratic view of human motivations and preoccupations. Not all of his successors have retained the

preoccupations, but his concept of 'alienation' has stood. This is a situation where a person plays a role instead of behaving 'authentically', and is therefore divorced, alienated from, his comrades, who are also playing roles. Eventually the exterior, bogus reality becomes the world-view of the person, such that he becomes alienated from himself and can no longer identify and address his own desires and concerns.

Soren Kierkegaard was worried about how we can know anything at all in the madness that surrounds us, and created the philosophical position of existentialism, where the value and meaning of an act can only be evaluated by the actor, based on the information to hand. This kind of social relativity certainly decouples the individual from the group, in which condition self-censorship from mapping may be avoidable, and one can wear black and brood a lot. It does however insidiously suggest that there is no such thing as objective, external reality (or if there is it doesn't matter because no-one knows what it is). This is liable to abuse, because it means that standing on the corner and making faces is as worthwhile an occupation as relieving terrible suffering or building houses, if the idiot says it is. This aspect of existentialism is contradicted by the mapper experience, which leads mappers to believe that there is an external reality, of great subtlety, and although none of us has yet appreciated it in all its wonder, if one of us discovers a phenomenon, it will eventually prove compatible with any other phenomena we have discovered. In this sense, the external reality is important even if it is not perceivable.

Kierkegaard was followed by Jean-Paul Sartre who wrote of the condition of the members of a society that denies itself, and RD Laing, who took existentialist ideas into psychiatry, where he saw whole families colluding in maintaining one individual who had been identified as 'schizophrenic' is a condition of complete confusion, as they spent a significant amount of their resources, both financial and lifetime, in protecting the packer reality of their less than happy families against the threat of the mapper that has appeared in their midst. From the mapper point of view, the 'patient' is in the middle of a complex web of mystification and coercion, distributed amongst the whole family which must be deconstructed if all are to find happiness. The packer view is that Laing is 'blaming' the parents for 'causing the illness'. This means that Laing's work has fallen out of favour in a clinical situation, where the effective power is in the hands of the patient's relatives (or they wouldn't be a patient). However, Laing's colleague Melanie Klein, who inspired many of his own ideas, had worked in the industrial sector, and existentialist ideas succeeding Klein are still of interest in industrial psychology.

Recently Peter Senge of the Sloan Business School at MIT has been writing about Systems Thinking, which is an approach to problem solving based in forming mental models and taking account of things like feedback.

When we set out to understand why some people are so good at programming, we knew that the answer would be interesting, but we never expected to come up with a simple model that could also draw a unifying theme between so many mystical and philosophical schools. It is probably valuable to have done so, because with so many apparently different ways of saying the same thing kicking around, the situation for anyone trying to break out of packer thinking but not realising that stopping stopping yourself and learning some disciplines is the way to go, is very confusing. One's friends might even think one had turned into a wierdo!

## Mapping and ADHD

There is said to be a disease called Attention Deficit Hyperactivity Disorder (ADHD), which afflicts 3% of the population. Its sufferers can expect to have a difficult life, handicapped as they are, but with appropriate drugs and support, they can hope for some integration into society.

In terms of the mapper/packer model, we suspect that ADHD may just be the results of natural mapper children, effectively being much smarter than their peers, getting into worse and worse standoffs with the packers surrounding them, as they think harder and harder, trying to understand what the packer teachers, peers and relatives around them want of them, while the adults see the children as disobedient or diseased because they do not evidence the necessary dysfunction required to sit repetitively performing the same simple, pointless, rote activities while not behaving like a herd animal.

# How The Approach Developed

The development of this work has in itself an exercise in mapping, so it will be illustrative to describe how the picture came together.

The work was motivated by watching what happened as ISO 9001 was rolled out within the computing industry. It seemed that at best, it ensured that we could be confident that an ISO 9001-certified organisation was at least off the `Laurel and Hardy' level, where one is capable of losing the source code of the programs one's customers are running, but did nothing positive to improve the programming skills of the people doing programming within the industry. There was an incident some years ago where the employees of an organisation that provided software to drive giant flour mills had to visit a customer's site on a pretext, pull a ROM and copy it before disassembling the contents and maintaining the program. No-one that has ever been in such a situation will ever forget it. So ISO 9001 was good, but the real work needed to go into the `engineering judgement' and `common sense' referred to all over the best process documents - the bits we couldn't get by apeing car factories.

But then we saw that in some organisations, there was an unexamined but almost religious faith that by reducing everything to simplistic proceduralism perfection would be attained, and that the metres of shelfware comprised the necessary simple procedures. With the process around, the limited thinking that had been going on could be abandoned or better, stamped out, and everyone could run around being `professional' without actually achieving anything at all. In the old days, at least poor organisations actually had the source long enough to sell it to the customer and pay the rent!

We needed to find out what real programming is all about, to counter the negative effects of badly applied ISO 9001 as well as an important ingredient supplementing well applied ISO 9001. On the basis that there was something missing from the ISO 9001 description of the workplace, and in honour of the surrealistic London Underground announcement, the working title at that point was `Mind the Gap'.

We started with the observation that there are some programmers who are much better than most, and that they agree amongst themselves on who they are. They can talk amongst each other about programming, and although they often disagreed about value judgements, they often agreed on a great deal.

Of course, right from the beginning, we had trouble describing what we saw talking to great programmers, in `management speak'. We spent a long time arguing around in circles, trying to get a two-dimensional creature into the third dimension, by showing it a series of steps each smaller than the last. The whole notion was of course flawed, because no matter how thin one slices the step, it is still a three dimensional object, inaccessible to a two dimensional creature. But we didn't know that then.

At the same time, we were looking at the great programmers' mind-set from within, deconstructing our own mentation while working, and watching others. This made much better progress, and we identified the `Artisan Programmer' as a figure more like a craftsman of old than a modern production line worker very quickly.

We were also interested in the underlying cognitive neuropsychology of the programming act, but hardly got anywhere at all. We could not find much work tying subjective experience to its platform, and one of the areas we would have particularly liked to have looked at, gender differences in programming, seemed particularly sparsely covered. Neuropsychologists commented to us privately that cognitive gender differences are sometimes hard to research because of political `correctness' considerations in grant applications. However, in the absence of useful psychological research, we did attempt to construct an operational definition of a subjective experience. This is what eventually produced the one bit program thought experiment, which demolished the external process view utterly, and left us to concentrate on subjective experience.

Between Spring 1992 and Autumn 1995 we spent our time talking to programmers, and discussing and contemplating what we had learned. We must have tried hundreds of ways of `telling the story', and every one of them died on the language barrier. However, we had discovered that the same few issues kept coming up over and over again, on site after site, and these issues had right answers. These have been included as Design Principles. We had also discovered that there were some ideas and stories that we had gathered from great programmers that had a very positive effect on the novices we told them to. This material has also been included in the Stone.

Then in autumn 1995, Frederick W Kantor's extraordinary work of physics, *Information Mechanics* provided a major inspiration. In it, Kantor throws away all crutches and attempts to build a consistent picture of physics purely out of information concepts. Perhaps the solution to our problem would be to throw out all the language we knew didn't work, and use the language we knew did work. Perhaps through this kind of ontological rigour, we could construct a self-consistent picture, even if it was divorced from 'mainstream' reality, that we could at least see clearly.

Very quickly we focussed on the movement of consciousness, and saw the link to alchemy. Links to other mystical traditions followed quickly, and we tried using mystically inspired language to novices, and explained about the circularity of hermetic journeys. We found we could improve the performance of programmers better than ever before, but we still couldn't explain why in mainstream language. By now we were calling the project 'Deployed Consciousness'.

In summer 1997 we were pointed to ADHD, and immediately recognised in the character profiles of ADHD children, the great programmers we had been talking to. We could see what the kids were doing, but it seemed pretty obvious that the psychologists and other professionals dealing with them could not, or they would by teaching them real stuff like number theory instead of burning them out by prescribing amphetamines so that they sat down and performed mindless, packer school 'work'. This was a great shock, but it gave us an important clue: there really must be some kind of cognitive blindness that meant that the psychologists were simply unable to understand the kids, and couldn't even realise that there was something going on that they couldn't understand.

This showed us why the language problem existed - amazing though it seemed, we had to conclude that our colleagues really were all in one of two (and only two) possible states, and we could describe the differences between them. We quickly wrote this up, assuming some kind of underlying black-box neurology, possibly involving a shift in processing strategy once some resource or other reached a critical level and made a strategy switch optimal, and distributed it to friends that had been talking with us about this subject from the beginning.

We received a lot of feedback, most positive, but one comment proved critical. We were asked if there might be any tie-in between this work and ME (aka CFIDS), the debilitating post-viral disorder that smashes the lives of so many active, creative people. Many mappers seem to know several people who had suffered from ME, and we made a list. Yes, they were all energetic thinking people, and not the brash, anti-intellectual yuppies that were characterised as getting 'yuppie flu'. But further, they were all thinking people whose essentially gentle personalities led them to respond to acts of gross stupidity thrown with all the contempt a packer can muster with sadness, rather than say, anger or contempt. Poke a monkey with a stick for long enough and its hair will fall out. This is a physiological effect of sustained psychological cruelty. ME had appeared during a period when packer fundamentalism had broken out all over the developed world, leading to enormous amounts of stupidity and cruelty. ME might well be an effect. But why just the gentle ones? They were all highly active, being the sort that would retille the barn because it was sunny, or cycle across Canada to celebrate their recovery, although they were all daydreamers. Daydreaming couldn't be it anyway, because we are all daydreamers... and the penny dropped.

The difference between packers and mappers is that packers have been socially conditioned to suppress their natural faculty for building mental models by daydreaming, and fall back on rote learning procedural, action oriented responses instead. We could throw away the neurological black boxes, and just say 'daydreaming' to make the bridge to mainstream language. Then the empirical work, as well as the understanding of the nature of the language problem, all fitted into place.

And that was the journey of exploration we ended up taking. When we started we didn't know what we would find, but we felt sure it would be worth it. For the first three and a half years we managed to help a few novices develop but didn't seem to achieve much else. We were gathering material and looking for patterns.

The overall work took nearly six years, but that is good going for a deep result. If we had never started we would not have reached the end of our journey, which we can now offer to you to read much more quickly than that!

## Complexity Cosmology

It is the repeated experience of mappers that their high investment in a cognitive strategy that they do not know will pay off is usually worth it? Why is this? Do we have any pointers to a deep answer to this question?

One possibility lies in the way our universe seems to have a thing about complexity. We know that from the earliest

moment, structure has been emerging in the universe. We know that the physical constants in nature are just right for making atoms, stars, planets, complex chemicals. We haven't proven that the emergence of life was inevitable in the universe, but as we all know by now, put just about anything in a bucket and kick it the right way, and you'll get self-organisation.

We might take the approach of extending the ideas of Teilhard de Chardin and Vernor Vinge discussed earlier, and wonder if our behaviour in adding complexity to the universe by writing software is just evolution, operating at a cosmic scale, upping the rate of change again. Then we might say that we get to win from two directions - first because the complexity we see has been built up out of simpler layers, so by drawing our arbitrary system boundaries we will often find opportunities for complexity cancellation within those boundaries, and second, because by adding more complexity we are just doing what comes naturally.

Building complexity might be a natural arrow of time quite as much as entropy, and thus inherently achievable in this universe, for reasons that we do not yet understand. Quite where this is heading we don't know, but perhaps we have the chance to find out before we get there.

## The Prisoners' Dilemma, Freeware and Trust

The Prisoners' Dilemma was extensively studied as a model of first strike nuclear ballistic missile strategy. In it, two prisoners are held separately, and both are offered the following deal, 'If neither of you confess, you shall both go free. If both of you confess, you will both receive long sentences. If only one of you confesses, that one will receive a short sentence, but the other will receive a double long one.'

The thing is, unless I can be certain that you won't confess, the best thing I can do is to confess, and settle for a short or long sentence, but avoiding the double long one. You feel the same way. So unless we are both certain (remember the old packer 'certainty'), which we cannot be, we both end up with long sentences where we could have got off with none at all.

This result was depressing during the Cold War, when considerable strategic advantage could be gained from a first strike. While the game theorists insisted that a double launch was inevitable, the human race, faced with utter destruction, was able to behave rationally and avoid any kind of nuclear exchange at all, let alone the Spasm predicted by game theory.

What went wrong with game theory? It comes back to the problem of establishing the certainty, in packer terms, of the other's certainty that you are certain... it's just too difficult. In order to do it, both players must be what the theories called 'super-rational' - able to be both rational in themselves, and rational about the rationality of the other. There didn't seem to be any obvious way to achieve this except quoting Ghandi to people, which didn't seem too certain to people who were themselves armed with nukes.

Within the mapper/packer model, things are much easier. If we are both packers, we both get long sentences. If you are a packer, I must confess, because you will. But if we know each other, it is easy for us to recognise each other's ability to construct mental maps and reach direct, intimate understandings of them. If I know you are a mapper, I know you'll be able to figure the trick of the Dilemma, because it's not a big map. You know I'm a mapper so you will also be able to predict my full understanding of this simple trick. So we walk away. In other words, we win because the difference between idiots and sensible people is a discontinuous thing, but only visible to sensible people. To packers it's a gradation between the insane (mappers who can't think properly and so... er... escape), through people who can only memorise a few knowledge packets and so are stupid, to Responsible Persons who apply knowledge packets with robotic precision. Remarkably, although we will kill millions and waste vast amounts of wealth playing face-saving games to keep our noses properly in the air, as a species we held off from blowing up the planet. Perhaps it was something to do with there not being anyone left to impress...

Packer preoccupations with certainty without the vision to get it, coupled with the zero-sum game of material economics and scarcity, lead to a very constrained set of transactions that are possible. To do software engineering we must be mappers, and the Prisoners' Dilemma shows us that we have opportunities for seeing effective strategies denied to packers. Producing software is a non-zero-sum game - if I copy your program we both have it. And we are out of scarcity, if only because programming is well paid. So there are more kinds of transactions open to us than to any other group in history. We are already starting to see examples in the shared production of standards, commercial organisations placing key source in the public domain, and in the growth of the freeware market. The people doing these things are not doing poorly

out of it - the benefits of leadership often outweigh the costs of giving something away that you've still got anyway. Sound business judgement consists of correctly evaluating this new business environment, and unless we do this, we will incur opportunity costs while competitor organisations are doing it for themselves.

The software market is liable to remain interesting for quite some time!

## Predeterminism

In *The Structure of Scientific Revolutions*, Thomas Kuhn introduced the concept of a paradigm - an underlying theory of the world that one doesn't even recognise as a theory but instead calls 'reality'. Whole societies share paradigms, and they can have an extraordinary effect on the behaviour of a society's members. There was once a philosophical paradigm called 'predeterminism'. It said that God had everyone's life planned out at birth, and the trials that one was subjected to could not be avoided because they were the will of God and so must be borne with good grace. Then there was a religious debate that ended up with the point that this contradicted free will winning out, and predeterminism bit the dust. This was good news, because the thing about predeterminism is that people don't do much. If everything is down to God's will, our puny efforts won't count for much.

With predeterminism out of the way, we were free to believe we could have some control over our fate, and so we did.

Ever since then, we've been waiting for the other shoe to drop. Although we believe that results are possible, and so we make efforts to better our lives, most of us still don't believe that understanding is possible, so we don't make efforts to understand. Now that our automation has both made understanding necessary and proved it possible, we have an opportunity to enter a new age of human experience - the true Information Age.

*This file last updated 10 November 1997*

*Copyright (c) Alan G Carter and Colston Sanger 1997*

[alan@melloworld.com](mailto:alan@melloworld.com)

[colston@shotters.dircon.co.uk](mailto:colston@shotters.dircon.co.uk)