Etched in Stone
Some things Teachers should know about Computers

Stuart Rushton
Lorien Novalis R.Steiner School,
NSW., Australia
stuart@lorien.com.au

Numberless are the world's wonders, but none
More wonderful than man; the storm grey sea
Yields to his prows, the huge crests bear him high;
Earth, holy and inexhaustible, is graven
With shining furrows where his plows have gone
Year after year, the timeless labour of stallions......

Nanoscopic petroglyphs, fifty million of them, etched in stone and connected by filaments of copper, gold and silver. Electric charges skip across the mosaic of petroglyphs, at close to the speed of light and in a path predetermined by the etched forms. The electric charges can be switched on and off up to 3 000 000 000 times every second. And its all happening on a piece of crystal the size of my thumbnail. If I could ask the crystal a complex question, one solvable by logical process, and in a language it understood, I would get an answer almost instantaneously. I am speaking about the Central Processing Unit (CPU) of my computer.

I have had a close relationship to cutting-edge technology and the ethics of its application for all of my adult life, firstly, for twenty years to the mid-1980s as an RAAF pilot, then for twenty years as a class teacher and teacher of science and technology at Lorien Novalis School. But of all technology, of every shining furrow we have created with our amazing skill, computers stand apart.

Steiner teachers have, as a principle, helped students achieve a creative and ethical relationship to technology. So why is computer technology proving so difficult?

During the late '60s through to 1980, I lived and worked for long periods of time in Papua New Guinea. It was during that time that I experienced the second wave of Cargo Cultism, the first wave started in the 17th Century with the arrival of "ancestral big men", in tall sailing ships filled of gifts of the gods. On one occasion, after flying to a remote mountain landing strip to unload a tractor, trailer and medical supplies for the local administrator, my crew and I found our aircraft surrounded by hundreds of men in full ceremonial dress – but this was no sing sing. People, some of whom had walked for days, were there to see the bikpela grin balus bearing gifts from the gods, and the big men who flew it. It would appear that the juxta-positioning of materialism, mysterious technology and spiritual striving, can lead to quite irrational behaviour. Is the West experiencing computer inspired Cargo Cult? Quite possibly.
At any rate, teachers are genuinely concerned about the worldwide and almost blind acceptance of the pedagogical value of computer use from pre-school and up.

The main issues include occupational health and safety aspects of computer use, which include eye strain and repetitive strain injury; along with the brain-retarding effects of computers on young children through linear thinking, two dimensional imaging and homogenising processes generally. Add to this the complexity and rapid evolution of computer technology and it isn't surprising that for teachers, caution is the order of the day.

Take a step back. What is technology?

Technology, simply put, is something that helps us transform, transport or store; energy or matter – a hammer, a car or the mains electricity grid are all examples of technology. Computers, as a subset, form part of what is termed Information and Communications Technology (ICT). In this sphere of technology (which includes pencils and quills), what is being transformed, transported and stored is in the realm of the mind - ideas, texts, images and information as simple as a phone number or as complex as the dynamic model for global weather, all of which is referred to, over simplistically, as data.

In this article I will first outline two important ideas that lie outside the usual computers in Steiner education discourse, namely, a challenge for all teachers to develop a right relationship to computer hardware. The second idea involves some deeper questions regarding computer software and critical technological literacy. Finally, I will outline how we are currently working with computers at Lorien Novalis School.

Andragogy and Pedagogy - learning what makes a computer tick

The lightboned birds and beasts that cling to cover,
The lithe fish lighting their reaches of dim water,
All are taken, tamed in the net of his mind;
The lion on the hill, the wild horse windy-maned,
Resign to him;and his blunt yoke has broken
The sultry shoulders of the mountain bull......

Is it important that teachers, including those who will never teach using a computer or even use a computer privately to have a right relationship to computer technology? I think so.

The journey we take as teachers to continue our self development – our andragogy – is very important. Part of that journey is to develop an understanding of the world as it impacts on our students, as such, computer technology is a phenomenon requiring a deep understanding. Computers have the impact of a fast flowing river, exciting and dangerous, and students of all ages are experiencing the adventure and seeking answers from their teachers. We must at least acknowledge and respect computers as part of human evolution.

During my seven years as a class teacher I constantly drew on world texts for imaginative insight, such as Sophocles' chorus above. I consider it the grist of class teaching to be in a world filled with lithe fish and lightboned birds. Seven years went by in the wink of an eye, all without a computer in sight. And thats the way I think it should be.
However, my class knew that I sometimes used computers in my work, that I knew interesting things about computers, they knew for instance that Lord Byron's sister Ada Lovelace, was the world's first computer programmer, and they had developed a lot of skills that would be easily translated into computer skills in high school.

My deeper understanding of computers is a work in progress which leads me, among other things, to search for the task of the silica in the world. Silica, the most abundant mineral on earth (sometimes confused with the element silicon or the lubricant/sealant/prosthetic compound silicone) is also present in our physical bodies as, for example, silicic acid in our skin, hair and eyes. Through spiritual science I strive to understand silica as generalised outer perception and see its relationship to Aires, the human head and the sound of the letter V. I note its hexagonal crystal form and think of Artemes and the queen bee. I wonder about the task of computers in human life. At how a technology more rigid than the human skull, and as far removed from mediating, healing breath as any process can be, will eventually play out.

I have learnt that at the core of every computer is a silicon crystal, extracted from silica - usually in the form of beach sand - and grown, as one grows a salt or sugar crystal.
Further, that our physical world is made of three classes of elements: metals, non-metals and semi-metals. Computers must use all three classes.
Firstly the metals – the electrical conductors. These allow electricity to flow though a computer's structures, wires and connections; made from iron, copper, mercury, lead, tin, gold and silver as well as other metals – in every computer.
Secondly the non-metals – the electrical insulators - usually in the form of organic compounds like rubber (natural) or polyvinyls (synthetic). These materials restrict the flow of electricity, guiding it into a complex matrix of conducting metals.
The **third c**lass of substances in every computer are the semi-metals – the semi-conductors. Here silicon with the addition of arsenic or other elements in homeopathic proportions can be made to switch the electricity on and off very rapidly and in predictable ways.

On face value, the wire, insulation and switch described above, looks a bit unremarkable, until we consider the scale at which the silicon switch operates. A two square centimetre silicon chip might be etched with a billion switches (uniquely interconnected), being switched on and off a billion times each second (in a specific sequence). It gives me a headache just thinking about it.

What I have just been marveling - and encouraging teachers generally to wonder about – is the physical aspect of computers, what is referred to as computer hardware. It is from one point of view a analogue for physicality of a human head, devoid of spirit – egoless. Without further human input the hardware will remain inert – it only has the potential to be technology.

Critical Technological Literacy

Words also, and thought as rapid as air,
He fashions to his good use; statecraft is his,
And his the skill that deflects the arrows of snow,
The spears of winter rain: from every wind
He has made himself secure – from all but one:
In the late wind of death he cannot stand.....

In whatever area we teach, we all strive for our students to develop *critical literacy* skills. To look beyond the obvious, to strive to understand the manifold nature of any given text, to look for the hidden curriculum – the value-ladenness. Similarly *critical technological literacy* has come to mean the same nous in relationship to electronic media and in particular the Internet. While these themes are well covered in a general sense, there is an deeper aspect of critical technological literacy that is seldom addressed – the realm of computer languages, in other words, the software.

At the level of the silicon crystal, the software consists of a stream of ONs and OFFs. This is generally expressed in binary form as ones and zeros.

0010100100010111110101000100100000
10100101010101001010111101010010101
0101010100101000010010100101001111
1010010100101001010100101001001010
10010

A couple of language levels up and the zeros and ones appear as programming instructions – here part of a program to print a sine table in the programming language C.

```
#include < stdio.h>
#include < math.h>

void main()
{
   int    angle_degree;
   double angle_radian, pi, value;

              /* Print a header */
   printf ("\nCompute a table of the sine
             function\n\n");

           /* obtain pi once for all */
   /* or just use pi = M_PI, where
             M_PI is defined in math.h    */

  pi = 4.0*atan(1.0);
printf ( " Value of PI = %f \n\n", pi );
```

All of this is invisible to the computer user - thank goodness I hear you sigh! However, it will help if you imagine, while reading the next few paragraphs, that you are fluent at reading and writing in a programming language (as many people today are).

Choosing a computer software model for schools.

There are two competing paradigms for the production and ownership of computer code. One is the proprietary or closed source which includes Microsoft Windows and Macintosh operating systems and applications such as Microsoft Office and Adobe Photoshop. Proprietary software remains the property of the company which made it and is characterised by quality products, expensive on-going licences, good stability and security and closed code (the programming instructions, in C for example, are secret and it is illegal to access or change them in any way), one-size-fits-all design and product support for the life of the licence.

The other software paradigm is open source software which includes operating systems such as GNU/Linux and FreeBSD as well as applications such as Open Office, and Gnu Image Manipulation Program (GIMP). Open source software is mostly in the public domain under General Public Licence (GPL) and is characterised by quality products, cheaper or no licence fees, excellent stability and security, open code (the programming instructions are available for all to read, copy and modify), flexible modular design and product support through websites, newsgroups and online manuals (it has a less structured support model than proprietary software but still effective).

All software is value-laden, on one level because it can be closed source - secret and hence unknown, or open source - available for critical analysis and pedagogical processes. Also, despite genuine attempts by companies like Microsoft to internationalise their software, the fact remains that it is a closed product, produced by a small pool of programmers in Redmond, Washington State, USA. Whereas, open source software is produced by a large world wide community of volunteer programmers.

The ethics of software.

The development of *open source software* and in particular Linux offers schools real choice. It has the hallmarks of a global social policy issue. As a form of globalisation "from below" it addresses concerns of fairness, ownership and monopoly.

Proprietary(closed source) software is in a bind with regard to the unethical/illegal copying of operating systems and applications. Their solution is to find stronger legal and technical means to recover the money they are owed. *Open source software*, on the other hand, offers free equivalents to the software thus overcoming the illegal practices but at the same time reducing the client base of the proprietary software vendors. Schools and universities provide "free" proprietary software to students for the duration of their learning. Graduates are then expected to spend thousands of dollars buying licences for what they have used for free since primary school – this is classical branding. Many graduates copy software illegally from a friend – it could be argued that current education practices create an environment for the committing of fraud by students. *Open source software* offers a legal free alternative to expensive software for students.

Many open source software solutions will work well on older computers thus extending the life of hardware and supporting the sustainable development objectives/policies of Steiner schools Open source software helps remove the barriers to computer literacy among the world's poor and disadvantaged by providing very inexpensive, flexible solutions.

After considering the issues and after discussions with our senior students we decided that our computer education should be based around open source software. The next step was to find a creative way of making it happen.

Computers: When and How

O clear intelligence, force beyond all measure
O fate of man, working both good and evil!
When the laws are kept, how proudly his city stands!
When the laws are broken, what of his city then?
Never may the anarchic man find rest at my hearth;
Never be it said that my thoughts are his thoughts.
Chorus from Antigone by Sophocles

It is impossible to argue, from my understanding of child development, that computers are an appropriate educational tool in primary school, indeed the idea seems to fly in the face of natural laws. There is already evidence of what happens to children when the laws are broken, nevertheless the global imperatives seem unstoppable - sell more computers, sell more software, for our Knowledge Nation (Australian education policy initiative) and so that No Child [is] Left Behind (USA education policy initiative) etc. If our children are not computer literate prior to birth, they will never get a job – hyperbole intended.

"When"is a hotly debated aspect of computers in education, as can be seen by my hyperbolic lapse above, but not one I will dwell on: except to point to *Fool's Gold: A Critical Look at Computers in Childhood and Children* and *Computers: a Call for Action* by the Alliance for Childhood as an excellent overview of the arguments (available at www.allianceforchildhood.com).

At Lorien Novalis School some teachers and senior students built a computer education resource starting in 2001 and totally independent from the school's administration computer network. Keeping it separate was our first important decision. It meant we could build a computer network free from the justified neurosis which accompanies having students on a schools main computer network. We could also design it, build it, break it, fix it and maintain it ourselves.

Our second important decision was to use Open Source Software rather than Microsoft Windows or Macintosh. The senior students involved were in favour of the Linux operating system – so Linux it was.
Choosing Linux meant we would save a lot of money on software licences, we could use older (recycled) computers and we could be part of a world community of volunteer computer software developers (in our case as users).

Our Linux Computer Lab is a work in progress and currently has 30 networked computer workstations for whole class lessons, and a few that are remotely located. There are a couple of file servers and an Internet gateway router, two print servers and printers, a scanner and a digital camera station. One computer has its screen projected onto a wall and is used by teachers and students show or demonstrate work.

Students from classes 8 to 10 have one hour per week of computer technology workshops. In addition computers are integrated into other subject lessons and some topics such as binary logic and computer hardware are covered in science, mathematics and technology main-lessons. Over classes eleven and twelve all students complete the 120 hour (some elect to complete 240 hours) VET Certificate II in Information Technology (IT), The VET IT course covers occupational health and safety, working in an IT environment as well as general skills in computer technology.

Our high school students develop skills

and an understanding of computers through studies in: computers and society, health and safety issues, file and knowledge management, computer hardware, software and ethics, word processing, web design and publishing including HTML scripting, building the school intranet, using the Internet, using a mail client, spreadsheets, presentations, graphics manipulation, databases, digital film editing, and computer networks.

Finally and most importantly, weekly computing in high school should be offset by weekly movement, breath and rhythmic activities such as Creative Speech, Eurythmy, music and dancing. Please feel free to contact us on the student network at linuxlab@lorien.com.au

Stuart Rushton is a teacher/administrator at Lorien Novalis School, Sydney. He has a deep interest in teacher education and in the history, philosophy and application of technology, especially in education. He is currently completing a PhD through Macquarie University with the working title Problematising Change: Steiner/ Waldorf Education and ICT.

This article was first published in *Towards the Future - R.Steiner High School Education.* Editors: Neil Carter, Ineke Mulder, 2005. Copies available from : waldorf@clear.net.nz