Consultation on 'National Policy on ICTs in School Education'

April 29th-30th, 2008
New Delhi

Short Discussion Paper

What policy should India adopt for ICT in school education?

Nagarjuna G.
Free Software Foundation of India, Mumbai

What Policy Should India Adopt for ICT in School Education?

Draft, meant for wider circulation with a request for comments.

Author: Nagarjuna G. nagarjun@gnowledge.org

Date: 28/04/08

The sections below develop reasons for adoption of free software exclusively for school science education. Towards the end, a list of policy guidelines and recommendations are provided for discussion. The document does not cover the pros and cons of using ICT for education. I will try to intervene during the deliberations on these issues.

Why Free Software for education?

Digitization of cultural resources can facilitate the conduct of education, or for that matter any domains where ICT is being used or recommended, if and only if it is conducted in the lines of free software movement, else it is going to go against social and democratic values. I will elaborate my argument below.

The argument can be best understood if we focus on what happens when we digitize any document, whether text or other kinds of media. Digitization uses a computing model to write (encode) the data, and when we try to retrieve the data, the computer reads (decodes) it for us in a human readable form. Normally, we expect that the computing model used for encoding and decoding is part of computer science, and so we rely on it.

However, since code is by nature arbitrary, each company can invent (mind you an arbitrary invention is not necessarily an innovation) its own model of digitization and provides a computing service to its customers. The arbitrary computing model they use is protected under the various forms of IPR, and the current Governments not only respect this but also protect and promote the interests of the companies.

Free Software Movement (FSM) identifies this as the root cause of betrayal that happens in the digital society in various forms. The computing model used must be published, just as any scientific or technological models are published for use by the society. More important than providing access, by publication, is the freedom to use the computing model by other agencies. In addition to this, FSM also seeks the freedom to modify the model, as well as the freedom to republish the model either without any restrictions or with the restriction that other users cannot transform them into private property.

Considering that computer science is a strange mix of deep theory as well as sophisticated technology, it is very vital for any society to use this transparently. Else, we will let some agencies become monopolies. This is a serious danger to digital society because, the data that is digitized belongs to you and me, and not to the company. But, in reality today, our data has been handed over to the proprietary companies, since they alone have the license to decode our own documents. This will create a possibility for computer crime, which is happening all around our eyes. We should not let this happen?

The only way of stopping this to follow the guidelines provided by FSM to correct this seriousdefect in the current digital society.

All cultural resources are at stake

All of us know well that the process of education is an important agency of cultural transmission, whether formal or informal mode of education. In traditional (non-digital) society, this happened by using natural languages, whose encoding and decoding knowledge (syntax and semantics) is stored only in the "way of life". The language games we play naturally stores this knowledge, and therefore we can use these langauges in the process of education.

For formal languages (scientific and mathematical models), where in the syntax and semantics is artificially declared, used heavily in science and technology, the coding and decoding rules are publicly archived, and accessible as cultural resources to all the people.

What is so special about computer languages that we cannot continue to practice as we do in the case of natural langauges and formal languages? This is the serious question FSM raises, and asks all the policy makers and social engineers to take immediate notice of.

We can learn a lesson from the cave writings and art. The ancient humans inscribed them on a hard stone with the hope that their inscriptions are passed onto us. However, they forgot to add in their inscriptions how to decode them. Since that code is not used, we lost them completely, leaving all of us to decipher what they meant. Though the art of deciphering is an interesting engagement, we should not let that happen to our digital documents that were inscribed only a while ago.

Most of us think that, by keeping the CDs, hard disks, and taking backups of files in number of locations, we are safe. This is a myth. For this digital code is nothing but the caveman's stone. For most of us are taking backups of only the stone (code) and not the rules and the means of interpreting that code. The rules and means of interpreting the code that contained in our own writings are already appropriated by several companies. We hold onto our precious 'stones' thinking that they belong to us. What is the use of holding onto stone engravings if we have no means of reading them.

If we understand how digital reading and writing takes place, and if we understand how users are prevented from reading, and also understand how the companies encourage us to write and create more and more cultural resources using the privately appropriated means, what we are doing is nothing but pouring our cultural resources into their kitty, which is a black hole for us, and a property for them. This is how the current digital business is working. We are all made slaves of this new so called knowledge society. [I recommend, by the way, not to use the term, 'knowledge society', for human society has always been a knowledge society. We should on the other hand use the more appropriate term 'digital society', for that is what it is. What we therefore need is a digital commission and not a knowledge commission.]

The danger is that every cultural resource (text, audio, video, music, dance, drama, educational tools, governing policies, nothing is spared) is getting digitized, in the name of efficiency, speed, modernity, and what not. If the society cannot claim them back, if we do not take precautions, and follow the model of FSM, the human society is in great danger of loosing our precious memories.

All the hype that is created in the name of emerging, so called, knowledge society is a way of promoting this process, where the commons loose all the power.

What are the implications of this picture for the use of ICT in education?

It helps us to define what is a free (mukta) digital learning resource (digitized culture) and how to protect it from private appropriation.

It has been a practice to distinguish the learning resources (often called content) from the technology (the software) and infra-structure. This is a serious mistake, and the current drafts that are circulated by the MHRD commit this mistake. This dichotomy promotes private appropriation. How?

Having made the point clear in the above section, that the use of a digital resource lies in the ability to decode the 'content' (code), and not in having access or possession to it, it should be clear why we must not separate the two. Interpretation (decoding) is performed by the software or the embedded software/logic in the hardware. If the model of decoding is not public knowledge, and even if there exists 'open' access to content, our lives are continued to be appropriated by the private agencies. That is why open access and open content is not enough, we need mukta knowledge.

A public cultural resource, which is otherwise eminently imitatable, can be commodified (for want of a better word) by technically separating the code from the decoding mechanism. Best example is a digitized song, or a CBT material available to us in the form of e.g. a CD. If you do not also have a particular operating system, these resources cannot be decoded. The company that makes the CBTs, or the songs have private agreements to twist the arms of the customers, and often they have several patents and other agencies must pay license fee to decode the songs and CBTs for us (At the end, we, as customers, pay for all this.). Together, the OS and software makers and the content creators are let loose, with support from Governemnts, to loot the people. [The current draft that is being circulated makes a special room for this in the name of PPP. PPP is not bad, but what service PP can provide and what not, and the rules for this must be clearly defined in the policy.] We must warn the Govt, and ask them to take immediate corrective steps.

In the case of mukta software, the scenario is different. The songs and the CBTs are encoded in a public model (for they use free and open standards), and the software that decodes and encodes is also made available always with source code. Disengaging mukta software with the source code is legally not permitted, for mukta software is protected by copyleft (GNU GPL or other such mukta licenses). Thus, by using copyleft model of legislation (policy) we can ensure that reading and writing are never disengaged from one another. This is how we can grant the freedom to read and write. I consider this right more important than the right to information, for what is the use of access to information without an ability to read.

If we do not take care of this, digitization will have a divastating effect on society. We will be dumping all cultural resources that carry our memories will be lost.

Considering that education is a part of the process of facilitating the cultural transmission of past memories (mimetics), we must not allow this digital devastation to happen. Unless, of course, we pledge to use exclusively mukta software.

How does copyleft work?

To understand how FS works, it is necessary to see how copyright can be used for creative works we write, including software. All of us are aware of the copyright law, and how most people use it. For example, a book published by MIT press, titled *Philosophy of Computing Information* has this on the copyright page:

> © 2004 by Blackwell Publishing Ltd
>
> All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, except as permitted by the UK Copyright, Designs, and Patents Act 1988, without prior permission of the publisher.

Many of us have seen such statements in most books. The copyright law says, the author or the copyright holder can specify the conditions of using the resource. But, as the history of its usage suggests, almost everyone in the world used it to restrict the way the resource can be or cannot be used by others. Only one person thought differently about this copyright act. He is Richard M. Stallman (popularly known as RMS). He began to invent different ways of writing a copyright statement. For example the book titled *Free Software Free Society*, that contains selected essays of RMS has this copyright statement:

> ©2002 Free Software Foundation
>
> Permission is granted to make and distribute verbatim copies of this book provided the copyright notice and this permission notice are preserved on all copies.

This was a creative use of the copyright. By imposing the condition, "provided the copyright notice and this permission notice are preserved on all copies", the author can ensure that only unrestricted copying of the work can take place. Else someone can reimpose restrictions. This line of thinking is the beginning of a new era of publishing not only written materials but also software. Though the above copyright statement shows 2002 as the year, RMS invented this art around 1983. Such a creative act of exploring a new possibility when other people don't see any, was called a hack. Around that time it was popular among the computer scientists to use the term 'hacking' for this kind of use. From that time, whenever someone finds an intelligent way of getting around a problem, it was called hacking. Popular usage by the media of the same term mostly means unauthorized intrusion by somebody into your computer, often causing damage or stealing information etc. For this kind of evil act, free software community has a different term, it is called 'cracking'. When we say, FS like GNU or BSD or Linux is made by hackers, we meant the former creative people, not the latter criminals. Cracking is a criminal act, but not hacking. Please therefore don't popularize the term hacking for the negative connotation, use cracking for this purpose instead. I found such terms in several serious documents, that is why this clarification.

Thus the greatness of RMS is not that he is one of the greatest programmers, though he is indeed one. History will remember him for the most important invention, nowadays aptly called, copyleft. A copyleft is an intelligent use of copyright for giving and protecting freedom of using a cultural resource, than imposing restrictions. RMS invented GPL (General Public Innocence) for this purpose. It is mostly used for software. GPL is a long carefully crafted document, a must read to understand the spirit of free software movement. You can find it in any copy of such software or better read it from http://www.gnu.org/copyleft/gpl.html, among other interesting essays about free software movement.

Definition of a free digital lerning resource

Along the lines of how FSM protects freedom, I propose here a definition of what can be called a free digital learning resource.

A free (mukta) digital learning resource (FDLR) must meet the following criteria:

- the users have the freedom to use the resource for any purpose, this includes commercial, non-educational or recreational purpose.
- the users must have the freedom to interpret/decode the FDLR. This is to ensure that the content is not locked in a proprietary format, and no licensing fee should be levied for reading or using the resources.
- the users must have the freedom to make changes, adapt to their context of the FDLR
- the users must also have the freedom to distribute the modified FDLR to the rest of the world.

FSM is not about spreading idealism

It is the most successful amalgam of action guided by a strong philosophical foundation.

Swatantra software model is today used by millions all over the world. The story of the evolution of free software is the story of a revolution that is still happening in front of our eyes. Very few revolutions can be witnessed this way. More than 80% of the Internet uses free software technology and free protocols. I do not want to go on listing the success stories and tell you that IITs use it, TIFR uses it, Google uses it, NASA uses it, it works on the space ships and satellites, it has been deployed in weather calculations and gene pool analysis etc. I almost started listing the success stories! But this listing is not what I intend to do share here. The point is to inform that FSM is not about promoting an idealism, it is praxis, action guided by a visionary manifesto. It is already demonstrated in substantial ways that it works.

We also know how many myths Wikipedia debunked? Today very few of us question the utility of such a collaborative way of creating socially useful resources.

Therefore, adoption of FS model for education in India cannot be questioned.

Educational resources, as well as the software and infrastructure, that we need in this country can be created using this model. This encourages, apart from the advantages of technical merits, a sharing culture, collaborively negotiating our lives and problems, promotes peer to peer culture in place of master to slave culture.

How Governments and other public bodies are helping Proprietary Software

We hear number of times in the news that a big software company signed a MOU with a Government, a school, or a college or an university. If we look deep into these MOU, we see some clauses, that make these organizations to use that software produced by that company exclusively. If it was a school or college, this meant teaching only the companies' software to the students. As a result, what we see are brand names in the syllabus as well as text books. For example, visit the website of the most popular open university in India. Their courses train students and examine their knowledge of only proprietary software. Instead of teaching generic software knowledge, they indulge in branded education. This is another serious crime committed by these public bodies. When asked, one state secretary told me that Government does not have enough funds to either buy computers or the software to run them. When companies come forward wearing philanthropic hat, Governments lay red carpet for them. Usually whenever such MOUs were signed, I was told, the companies gave away a number of licenses to schools either gratis or with a substantial reduction of the original price.

This act is nothing short of giving tobacco to small children gratis, till they get addicted, so that addicted people will begin to buy cigarettes. The Governments and public bodies must realize the evil intentions of the companies in doing these so called philanthropic gestures. The education department of the state may have saved some money, but other departments in the state will loose lots of money, for they need to import the software. Most importantly we lost our freedom. Our education system is actually spending money and most precious of all time, for the benefit of a multinational company. We must not loose our freedom this way.

FS is an eminent and indispensible choice for education

When a car breaks down on a high way, the driver opens the bonnet and checks if he can fix the problem. Often, experienced drivers do so. If he does not know, he will take the car to the garage or call a mechanic if he cannot tow it to the garage. The mechanic in the garage fixes the problem, and the car is back on the road. This is what happens with every other technology. Does this happen with ICT?

How many times your proprietary operating system yelled at you: "The program committed an illegal operation, please contact the program vendor" and quits from running the application. Several times, the OS hangs, displaying a blue screen. Most of you will reboot the system, some times reinstall the 'corrupted' application, and it is not uncommon to install the OS all over again. These steps are not same as repairing the system. If you wanted to repair, the first requirement is to obtain the source code. Even if you are one of the best programmers in the town, you will experience the misery. All the knowledge and training you had was useless. Even though you knew how to fix the problem, you could not. This is how an educated and well trained engineers are treated with proprietary software.

This does not happen with FS. When the program did not work, you could download the source code, look at the program, identify the bug, and possibly fix it. After fixing the problem, you write to the main developers, informing about your feat. They congratulate you, and then you become part of that team. This way, a knowledgeable user is invited to contribute to further the development process. Even if we are not knowledgeable, there exists a possibility to get the problem fixed from another geek in town, whom you could compensate by paying for his time.

I do not know a single concept in computer engineering that cannot be taught using FS exclusively. Since FS does not belong to any one company, the training will not lead to any monopoly. Since source code for all programs is accessible, students can learn by actually looking at the program, better still by modifying them. This is the way all other technologies are taught. But due to lack of a good policy with regard to IT training, IT education in our country, and also in several other parts of the world, got associated with learning a particular brand. All this learning is restricted to how to use, and never how to repair.

How can we make good engineers if they cannot be trained to repair? How can we repair a car, if the car comes with a closed bonnet? It is therefore very important to work towards a policy to teach ICT exclusively with FS.

Free Software and Open Source Software

I do not recommend the use of the term “open source”, since it is not a well defined concept, unlike the term 'free software'.

Most people think that 'open source' is clearer in meaning while 'free software' is confusing. This is a misconception propagated by the OSS advocates. The confusion is due to imposing a specific meaning from other uses of the term 'free', without realizing that 'free software' is a well defined term --- i.e., any software that has the four freedoms is a free software (elaborated below). This definition helps us to arrive at an unambiguous class of software that gives the users the four freedoms. The philosophy guiding the free software movement (FSM) is an exemplar par excellence of clear practical thinking.

What is this thing called free software?

The first thing we need to know about the term “free software” is that its meaning does not arise from the combination of the terms “free” and “software”. The meaning of this term arises from the definition, and not from the terms it contains. The term âfree softwareâ is defined by the Free Software Foundation (gnu.org) as that software which gives the user the freedom

1. to use it for any purpose,
2. to know how it works,
3. to improve it by modifying, and
4. to share or propagate or distribute the modified code to others.

Any software that meets these four criteria can be called free software (henceforth called FS). We must notice that there is no mention of price of software in its definition. This means that there exists a possibility to pay or charge for FS. Since FS is intended to give the users the freedoms mentioned above, it is better called as “freedom software”. In Indian languages there are more options: we may call it “swatantra software” (a preferred term in southern and western India), or call it “ajaadi” software (a preferred term in north-eastern India), or else call it “mukta software” (a preferred term in northern India). The last option is nice, since we can have pun with the word to say: we are talking about mukta, and not mufta (gratis) software. Let us therefore bear in mind that free software is not about price, but it is about freedom. Richard Stallman, who founded FSF and the GNU project, says aptly: “free software” is a matter of liberty, not price. To understand the concept, you should think of “free” as in “free speech,” not as in “free” [cup of tea].

Another way to make the issue clearer is: take the terms 'energy', 'work', 'force' as used in Physics. They all have a meaning in folklore. If we think they have the same meaning in science too as in folklore just because they are spelled similar, we are obviously wrong. The terms, when used in Physics are well defined technical terms stated by operational criteria. Similarly, 'free software' is re-defined concept though the two terms are taken from folklore. Just as scientists have accustomed to the re-defined concepts of 'work', 'force' etc. we may accept the term 'free software'. In an Indian context, we can safely use the term “mukta software”.

The term 'open source' on the other hand is defined by about ten criteria approved by OSI (open source initiative) (http://www.opensource.org/docs/osd). After looking at the criteria you will agree with me that the definition is free software is not only more elegant, and operational but also has a positive social message. A few of the licenses that are declared OSS (open source software) are not FS licenses. Therefore there is a slight difference in the reference too. Due to these reasons, OSS and FS are not same concepts. The term 'open source' is invented to make FS business friendly with the assumption that the ideology of freedom is not good for business. OSS advocates talk mostly about an efficient development methodology. Open source movement and free software movement are therefore not same movements. Time will prove that freedom is good for everyone interested in preserving human values, including ethical business.

Therefore, the terms 'free software' and 'open source' are very different in their meaning, but they refer to more or less the same set of software with negligible exceptions. The meaning of being open is implicit in freedom, for freedom is not possible without being open, but being open does not necessarily imply freedom. Thus, semantically FS is inclusive of openness. Most importantly, when we are framing a policy we need clearly and easily operationalizable criteria, so that it is easily decided.

Some people use the term 'freeware', assuming that FS is similar to it. But, freeware does not give any of the freedoms, but it is only free of charge. Another term 'shareware' is also used, but this is another name for proprietary software that is meant to be given free of charge for a trial period.

Some others misrepresent FS as 'public domain' software, without realizing that public domain software does not state clearly the conditions of usage, usually they have none. Public domain software is FS, but not vice versa.

It is therefore very important to realize that FS is different and novel from others. The only antecedent of FS philosophy in human history is good human values that nurtured our societies in the past and preserved the culture. FSM though is rooted in virtuous cultures of human kind, it is only now an overt attempt to transform the knowledge dynamics of society with a clearly stated manifesto got initiated. Do read the GNU manifesto from http://www.gnu.org/gnu/manifesto.html, to see that this is more an ethical and political movement than a technical movement. It is incidental that FS is also technically superior and economical.

Other documents to read

- Why schools should exclusively use free software
  http://www.gnu.org/philosophy/schools.html
- Philosophy of the GNU Project

  http://www.gnu.org/philosophy/philosophy.html

ICT Policy Recommendations from FSF India.

Guidelines

1. The right to digital encoding and decoding must be declared a fundamental right of the people and not the industry (currently the Governemnt grants this right only to the industry and not to the people).
2. The guide line for the ICT policy must be based on a fundamental principle: all cultural resources that are digitized must be readable (decodable) and writable (encodable) for eternity to ensure preservation of culture.
3. Digital encoding must be recognized as an extension of the social form of documenting various cultural forms like writing, publishing, and performing.

Specific Recommendations

- Documenting and Publishing
    1. All public digital encodings (data) must be a free and open standard.
    2. All public websites must be made interoperable, and must not publish its content in proprietary format (e.g., macromedia flash) or dependency requirements must not include special software. Public websites must not make their pages work only in one operating system. This violates inter-operability of the published content and services provided by the web sites.
    3. All indian language fonts which do not follow either ISCII or Unicode standard must be declared illegal. Font encoding (creating private mapping table between the font and the character) must be declared illegal. All current industries which are doing this must be warned and asked to comply to this within a stipulated time frame. Manufacturers should be made responsible for providing filters to convert all the existing documents without any additional expenditure, and such filters must be published as free software.
- ICT Education
    1. Technical education in general and ICT education in particular must focus on providing skills and not emphasize on brand names and products. This is already an implied policy based on MRTP Act, however technical education in both formal and informal sectors are not implementing these guidelines.
    2. Evaluation (examinations) of technical and ICT skills must not be based on a specific brand/product, but must test only the skills.
- Procurement of Software and Hardware
    1. Software that does not support free standards must not be baught, and used by a public body, and the software must warn the user if a proprietary encoding is used (this is contrary to what happens while saving a file in MS Office today, which warns the user that there will be loss if you save in an open standard like rtf or html or text). Applications should explicitly encourage the use of free and open standards, and discourage the use of proprietary standards.
    2. Importing hardware that does not publish its manufacturing specifications or do not provide free software drivers (software that controls the hardware) should be prohibited by law. Similar condition shuold apply to manufacturs in India.
- General Recommendations
    1. Govt should take initiateve in setting up of creation of an independent consortium to define from time to time what is a standard and what is not, what is best governance and what is not.

2. All ICT development projects funded by the people must be published without restrictions on their use, study, modifications and distribution. FSF India recommends the use of the appropriate copyleft licenses.
3. Public funds must be made available to develop free information infrastructures like, wikipedia, open access publications, knowledge bases, geographical information, maps etc.
4. Organizations representing free software community, such as Free Software Foundation of India must be considered as one of the core stake holders while deciding public ICT policy in the country, and the consultations must not be restricted to academic and industry bodies like NASSCOM, FICCI, and CII, who have not kept the people's interest, but served mostly the interests of the industry.

   Please send you comments to nagarjun@gnowledge.org or to the list.