The Importance of Open Standards

English version 0.2

Wim Deprez - wim@lumumba.luc.ac.be

4th November 2002

1 About this document

This document discusses the importance of open standards in the contemporary world of informatics and tries to clarify that educational policy can affect its future. In our society today, informatics and computer science are important. They have braught us technical and economical prosperity. Now that programmers start to cooperate massively, the need for open standards becomes more and more vital. Many people devote theirselves to make the information era as open and honest as possible, without turning it in a pointless money-tap. Unfortunately, most users are not aware of the inexhaustible source of possibilities and are rooted in certain systems that gained undeserved monopoly positions, at the cost of innovation by other products.

Since September the 10th 2002, this document is in version 0.2 and it is downloadable at http://lumumba.luc.ac.be/wim/OpenStandards/ along with other versions. Comments are always welcome at wim@lumumba.luc.ac.be .

Initially, this document is written in Dutch, this version is also available at http://lumumba.luc.ac.be/wim/OpenStandards/ .

2 What is an open standard?

Open source often gets compared to a pack of cornflakes, which mention what kind of ingredients they contain. Traditional *closed source* software can be compared to a white box, which only mentions the *license agreement.* That license agreement states that it is a violation to trace the ingredients of this food and that it is prohibited to try to copy the pack of cornflakes. The company just garantees that the stuff is edible and that they probably made the best cornflakes ever. And after purchasing the white box, you should of course buy the right brand of milk to go with it, else the flakes are not at their best. In our ”‘normal”’ world, something like that is impossible, even inconceivable, but at the other hand it should count for informatics? This statement is also applicable to *open standards*, the milk of software.

An open standard is a standard created for common use, the designers did not have any intention to make it profitable. Open standards can be used, implemented and their specifications can be read without fee or trouble with patents and copyrights. First of all, open standards are of course *open*, which means that in theory, everybody may cooperate at its development and that the standard may be used and read by everybody. Every participation is welcome (open meetings) and will be discussed if needed. There is no dominant authority who can impose its will on others, but there are ballots and debates so that the standard will reflect the public opinion. It will rather be governed and supervised by a impartial organ than dictated by a certain company.

A good, open standard should be like spoken languages, such as Dutch and English. It is free to use and usable for many purposes. Who would be such a fool to ask (no: to *demand*) a charge, every time a sentence is spoken, every time people just think about making a conversation. Spoken languages are used by everybody, for free, the standard itself is actually determined by the users. A restricted number of people is concerned with notating the prescriptions and make it an official standard, but the language itself stays quite flexible and extendable, new words can always be added. A spoken language can be implemented in many "‘products"’, among some are free (a conversation or a discussion) and others can be obtained in exchange for a small fee (a book, radio or television, ...).

The makers of open standard products have no preference for certain salesmen (or saleswomen). The products theirselves are characterized by a broad distribution of standards and are widely available, free or for a small fee.

It is important to remember that when something is used world-wide or is unattached to a specific operating system, it is not automaticly an open standard. The other way around may count. The world-wide use of *Adobe*'s *Portable Document Format* and the fact that it can be viewed at any operating system, does not make PDF an open standard.
An open standard has to be available without dificulties and for free or at a low price and its developers shouldn't be aiming for high profits. Open standards only prescribe how communication should happen or how the data should be represented. The choice of which application to use is up to the user. Thanks to the utilization of open standards, two differnt users can still work together and exchange data or set up a connection without the need of the same implementation.
A nice example is the use of the *JPEG*-format: although their are a countless number of different viewers and editors (from free to low-cost to prohibitive), people can open, view and edit a file that complies with the JPEG-specification in any supporting programme.

3 Qualities of open standards

To explain the qualities and principles of open standards, I would like to direct (with their approval) to the works of *Bruce Perens* [1] and *Ken Krechmer* [3].
As mentioned, an open standard first of all should be available to anyone to read *(open access)* or to implement. This happens mostly at websites where descriptions of the standard are hosted and where examples of implementations are free to download. The enclosed licenses may not exclude a party to use the standard as long as that party uses it as prescribed. Manuals are free downloadable or to buy for a small amount of money (not more expensive than a school book). Open standards create an honest, competitive market for its use: several applications should be possible, as well as for commercial, academic and personal use.
The organisation behind the standard may not discriminate, it may not prefer an implementor above another for any other reason than the observance of the prescriptions of the technical standards in their applications. An organisation that trades certificates, mostly distinguishes a standard price (mostly for commercial use) and a lower (or none at all) price (for academic or personal use).

An open standard may be expanded or divided. There should be condition to this to prevent implementors to create incompatibilities, it would dislocate the open standard. The licenses may not forbid expansions, but can oblige the implemantors to make a licence agreement by itself - by certain rules - which allow others to create, to distribute and to sell software that is compatible with the expansions.
This brings us to the so-called *Embrace & Enhance* technics: a seller puts an application on the market, that application contains extensions of the standard, which are incompatible with other systems that have implemented the standard too. If that seller has a dominant market share, then the orther systems are incompatible with the majority. Through patents and copyrights, the dominant seller restrains others of expanding their applications (so that those would be compatible), so those other companies will die a silent death because of the bigger market share of the dominant seller. This last one will obtain a monopoly position and the end users will be obliged to purchase its products to be compatible with the majority again. A good example is the clash for the largest market share between the different internet browsers.

With its *Sun Industry Standards Source License* [4], *Sun* has given an example to avoid the above. The Sun license commits the implementor who have designed an expansion to publish it. Other organisations for open standards could go on on this kind of license to preserve compatibility, without standing innovation in its way.

4 Why should we stimulate the use of open standards?

Companies who have a (as good as) monopoly position, have little motivation to enhance their products. The only thing that should never stop working, is the money tap, so they only have to keep their shareholders satisfied. They have no more urge for quality, as long as the products just keep selling. Companies only want to protect their profit, that is just the way our economy works. If you can puchase compatible products at only one specific firm, quality is no longer the first item on their agenda.
Open standards protect the right of choice of the consument, they create more posibilities and doesn't make him to toe the line concerning applications. As long as open standards are used, it doesn't matter what kind of implementation the user employes, because the standards are compatible, so people kan open, read and edit each others files.

If we observe the behaviour of the user of programmes that do not handle open standards, we see that when almost everybody uses the same product for a certain purpose, they also buy a new version at regular times, although it already was a respectable product and easy to use. Why is that necessary? It is the same as buying a new hammer and a new set of screwdrivers because they no longer fit in the workbench which get renewed every three years. Why does everybody keep upgrading and paying more for features *that they already have*?

As Doug Dingus [2] mentioned in his work, many users purchases the newest versions of such a product because they are just *compelled* to do that. Componies who have gained a monopoly position just stop with the sales of the previous version of their product and adapt the standard of the newest version a little. Because people experience problems when they exchange files between different versions, they are obliged to buy the newest version too. Of course, the new standard has learned some new trics, but most of them come only in handy for a small group of users and still the whole community gets punished and compelled to use the newest version.
This problem may not arise, different versions of standards should be able to operate next to each other. New implementations need to be able to handle older standards and the new versions of the standard need to be designed in such a way that they can be dealed with no (or the as less as possible) trouble in older programmes. If standards are designed like that, there is little *lock-in*, so that users are free to choose what version they want to use. Most open standards are implemented with this kind of *forward* and *backward* compatibility at the back of the designers mind, this is the true value for an application which does not want to be a flash in the pan.

Besides getting around dirty marketing strategies, open standards can also be *qualitatively* better. Because of the fact that the standard has no commercial significance for the designers, their is a possibility for *communication* and *participation* (open meetings). As such, designers can mix the best of each others ideas, nobody has to restrain himself to "'blab"'. Instead of a few whizz kids a company, people can *world-wide* put (as many of) their heads together.
Because there is so much perusal, there is so much been discussed and thought over, the first version of the standard will already support most features as good as possible. After a while, new versions will of course be published, since a standard is meant to be used a lot, people will gain experience with it and the bugs and unrounded corners will come out eventualy. It is just the open access that makes it possible for people to have a say and to bring the standard to perfection together (end users *and* whizz kids), step by step. This makes the debugging of the standard quite a lot goal-oriented and the standard itself more efficient and more secure (if we are for instance dealing with a standard for a communications protocol like the *HyperText Transfer Protocol (HTTP)*).

Open standards provides us also with better programmers and communities. Through the reduced number of standards, people can specialize better and establish organisations to help each other and to exchange experiences. Coders will work together in a better way because of the uniformity and will in this way complete projects of other people.

5 How should we stimulate the use of open standards?

In two words: ***Public Awareness***.

One should make clear to the users that they do *have* a choice, most consuments don't even have the faintest idea. As a computeruser, one has the *right* to a choice. This is very important and should start in the education. Instead of giving the choice and to teach informatics and its various branches in a general way, people are taught in the ways of a specific product. One should teach the essence and the principles of word processing in general instead of the use of a specific programme. If we want to give the user a broader perspective on informatics, we have to quit this "'product education"' and lay bare the alternatives. Most people are little motivated to sort it out on their own because they do not know the benefits.

Especially courses and presentations of high schools and universities which are free to download from the internet should keep to the standards and turn away of companies who apply the *Embrace & Enhance* technique in abundance.

If people during their education have learned to work with only one specific product, companies are obliged to purchase that product to. By the lack of motivation of the users theirselfs, the cost of retraining for the company and because education is not oriented on applications in general, people keep working with the same products for ever. This goes hand in hand with the monopoly position of those products and that is why there should be instructed in open and honest programmes of informatics in general during education. In this way, students will be motivated to make their own choice about which product to acquire. This is how education can help to undermine the present, wrong system. If educated employees know better applications, companies will know it soon too. This will be the beginning of an honest, more open and cheaper computer era.

6 What do the opponents of open standards say and are is that correct?

Just as any philosophy, reasoning and movement, the idea of open standards also gets some opposition and not only of the traditional software companies. The most commonly used argument [6] of people against the openness of standards, is that they would leave little room for innovation, but as mentioned above, this is most certainly not true. Openness really leaves room for innovation, even more and better than the traditional proprietary standards. The innovation is more supervised to prevent the *Embrace & Enhance* techniques. I refer to the *Sun Industry Standards Source License* [4].

The openness of standards also brings the advantage that they are easier to pass on to other platforms, to obtain the same functionality as on any other kind of platform. Think about the future, more and more new platforms are and will be developed, even for the television, all kinds of mobile telephones, PDA's, the refrigerator, the microwave oven, ... How would open standards hold back innovation?

7 Conclusion

Open standards are not meant to live *on*, but to *survive* in a world of communication and multimedia. Independent of the used programme and operating system, users should be able to open, read and edit mediafiles and to start communications and use protocols. This is how the end user keeps its individuality and its right to choose, without losing compatibility with other users (with their own individualities).

> *"'Nobody should be able to own your ability to perform basic computing tasks and make you continue paying for the ability to do so[2]."'.*

This is not a utopia, but an actual possibility. This is most certainly copeable, but it requires of the user to throw its blinkers away and education has to come to its aid. Only when the user seeks to it, monopoly positions will be knocked over. Yes, it will need a revolution, a software revolution.

8 Acknowledgements

This document came into being by the cooperation of Bruce Perens (bruce@perens.com) and Ken Krechmer (krechmer@csrstds.com), their splendid works on open standards (respectively [1] and [3]) and everybody of OpenStandaarden.be [5] by their constructive criticism in the mailinglist. Thanks to all of them.

I would like to thank Wim 'Alien' Stubbe (@) especially, because he introduced me to OpenStandaarden.be [5], my *spellingsnazis*[1] who pointed out my ”‘zpelllingzmiztakes”’ and all those who helped me to get the better ideas in and the lesser ones out.

References

[1] Bruce Perens, *Open Standards - Principles and Practice*, http://perens.com/OpenStandards/Definition.html

[2] Doug Dingus, *Why Open Standards Matter*, http://www.osopinion.com/perl/story/16159.html

[3] Ken Krechmer, *The Principles of Open Standards*, http://www.csrstds.com/openstds.html

[4] *Sun Industry Standards Source License*, http://www.openoffice.org/licenses/sissl_license.html

[5] *OpenStandaarden.be*, http://www.openstandaarden.be

Opposition (but do not forget to read through the *comments*)**:**

[6] Shelley Powers, *The Tyranny of Standards*, http://www.oreilly.com/news/tyranny_1200.html

[1] Have you spotted ”‘zpelllingzmiztakes”’ or (very) bad grammer? Become one of my *spellingsnazis* and mail me at wim@lumumba.luc.ac.be . Every additional comment is also always welcome. Thanks for taking the time to read my paper.