Proceedings of the 4th International Barcelona Conference on Higher Education

Vol. 2. Knowledge technologies for social transformation

GUNI - Global University Network for Innovation – www.guni-rmies.net

Preparing the Ne(x)t Generation: Lessons learnt from Free / Libre Open Source Software

Why *free* and *open* are pre-conditions and not options for higher education!

Andreas Meiszner
Researcher, The Open University
UK
&
Rüdiger Glott
Researcher, UNU-MERIT
Netherlands
&
Sulayman K. Sowe
Researcher, University of Thessaloniki
Greece

Quotation information

MEISZNER, Andreas; GLOTT, Rüdiger; SULAYMAN K., Sowe (2008), “Preparing the Ne(x)t Generation: Lessons learnt from Free / Libre Open Source Software”. *Proceedings of the 4th International Barcelona Conference on Higher Education, Vol. 2. Knowledge technologies for social transformation*. Barcelona: GUNI. Available at http://www.guni-rmies.net.

Abstract

1. Introduction
2. FLOSS communities as open participatory learning ecosystems
   - Openness and transparency: processes & structures
   - Freeness – Free as in free beer or free as ‘libre’
   - The power of the crowd: Critical mass and why size matters
   - Skills learnt in FLOSS – gaining soft skills as a side effect
3. The FLOSS production model and its applicability in HE
   - Including examples of FLOSS-like cases in educational settings
4. The FLOSS support model and its applicability in HE
   - Including examples of FLOSS-like cases in educational settings
5. The FLOSS business model and its applicability in HE
   - With reference to current OER projects
6. Conclusion

With reference to round table discussions on ‘preparing the new generation’ and ‘renewing knowledge creation’ this paper will illustrate why higher education (HE) needs to reposition itself to be prepared for the ne(x)t generation and which are the lessons to be learnt from well established virtual and informal open participatory learning ecosystems, in particular the Free / Libre Open and Source Software (FLOSS) communities.

As has become clear; FLOSS communities succeed in providing and distributing in a sustainable manner the knowledge necessary for the production of good quality software, thereby using a different development approach than proprietary software producers. One characteristic of FLOSS is that it is built by a community of volunteers and frequently backed by companies that generate their revenues by providing services related to FLOSS projects and the software.

In more recent years FLOSS communities also gained attention for their community production and support models and regarding their way of knowledge creation, sharing, and learning opportunities. Though FLOSS communities might be

the most mature learning ecosystems; the web is not short to provide further success cases that show how knowledge can be created collaboratively by armies of volunteers, how user to user support systems work, and how sustainability and quality can be assured through community involvement.

The coming ‘net generation’ is about to enter higher education. This generation not only grew up with ICT, but is also used to taking on an active role, to create and design resources, and to engage at the web with peers from all over the globe. This net generation is further used to two fundamentals of the so called web 2.0: ‘Open’ and ‘Free’. The web 2.0 intends to be ‘Open’, which means that in general there should be no access restrictions to participate at a given community; e.g. due to prior education, age, culture, or professional position. The net generation is further used to the freeness the web 2.0 features; both freeness of accessing content and communities (free in monetary terms), but also freeness in terms of freedom to express oneself and to be creative. The net generation is multiprocessing, multitasking and feels comfortable once navigating through the manifold information spaces, enjoying a mixture of learning and entertainment that might be perceived as a kind of infotainment.

In some regions of the world, like e.g. Extremadura in Spain, an entire generation of students is not only growing up as a net generation but additionally with open source software and open content, with the later being jointly produced and shared amongst teachers and students. Once those students will enter HE they will not only be ICT literate, but also have a mindset that demands for taking on active roles, to collaborate globally and to take change and modifications as granted.

So how will HE respond to those demands taking into consideration that the net generation is well aware about the options and alternatives the web provides? Challenges to HE in this respect are:

The FLOSS case, like also Wikipedia and the web at large, has shown that the sum is bigger than its parts. Acting in the virtual world challenges traditional laws: The knowledge is power rule, for example, only applies if knowledge is being shared with others, but not by ‘hoarding knowledge’ as this means to remain invisible; with the knowledge being provided by someone else. As a consequence ‘selling knowledge’ is equally difficult in the virtual world as someone else might be willing to provide the same knowledge for free.

In times where knowledge is becoming obsolete faster and faster, with a 4 years' university student having to face that half of what has been learned during the first year will be out of date by the third year of study, educational settings will need to adapt new structures and models to keep the pace. Education at large struggles to update their courses within shorter and shorter cycles or to develop new ones, with lessons still being largely given like 100 years ago.

One certainly could argue that HE can not be benchmarked with informal learning environments the web provides. After all, HE is a full service provider and offers recognized degrees that can not be compared to self studying at the web. However, the case of computer science education research has shown that 39% of surveyed IT companies (FLOSSPOLS survey) expressed that there is no difference between formal qualification and practical experience in FLOSS, with a further 17% claiming that formal qualification is even worse. This indicates that HE is well advised to keep an eye on the learning opportunities the web provides, especially in contexts where practical experience is considered equally or even more important than "theoretical" education at school or university.

Understanding web success cases like e.g. FLOSS is therefore crucial for HE to adapt itself to the new realities. The maybe most relevant characteristics of FLOSS communities, and partly also the web at large, that could help to improve (higher) education and to meet the net generation's expectations are likely:

1. The community production model
2. The community support model
3. The underlying business models to assure sustainability

1. The community production model

'Collaborative content creation', 're-use' and 'peer review' are key factors of the FLOSS production process and deeply embedded within the FLOSS community structures. On the other hand these factors are not systematically found within the educational landscape. Education systems are per se closed system where content is usually developed by the educational provider only and re-used or reviewed within a clear defined scenario (which is in general not a continuous process, but done 'once in a while'). Similarly the Open Educational Resource (OER) movement by now is still based on the traditional production model considering only educators and professionals.

2. The community support model

The FLOSS support model is already, through its conception of being provided on a voluntary base, very different from the, at least officially, guaranteed support system in educational settings. Personal support in FLOSS is in general provided through asynchronous communication tools such as mailing lists, forums or wikis; with the benefit of preserving the various answers and the process towards reaching the answers. Those recorded answers and documented processes are then becoming valuable learning resources for future learners, with individuals acting as knowledge brokers and point one the other to useful sources or leverage knowledge from one domain to another. Though synchronous (direct) personal support, as is typically provided in traditional educational settings, can be found too, but to a much lower degree. This support model provides risks as well as chances to learners. A risk is that learners must evaluate which source of information and support they can trust (there is no approval of teaching material by teachers or the like). Chances are a gain in information sources and the speed in which the community reacts to individual information demands, 24 hours / 7 days a week. The ubiquity of information resources and support is probably one of the strongest challenges for traditional HE.

3. The underlying business models to assure sustainability

The FLOSS case further provides an idea how revenues might be generated through providing services related to a product that is made available for free. FLOSS solutions, like e.g. Linux, are made freely available for everyone to use, modify and improve. However, there are underlying business models that allow companies to contribute to the development to the software without charge, but instead to generate services related to the software. Services might be provided to commercial companies for implementing the software or to train staff on using it. In the case of Linux for example DELL offer PCs with a pre-installed Linux Ubuntu distribution and allows

customers to choose between the freely available volunteer support system or alternatively paid support subscription plans. Looking beyond the current OER approaches those mixed business models might be an additional model for HE, as the knowledge that is taught and learnt is no longer a means for the production of goods that can be sold but the knowledge itself becomes a good.

The FLOSS case, but also the web at large, provides an insight on how HE might benefit from going a step beyond the current OER move, and shows which principles, structures and strategies could be adopted and which are the pitfalls to be avoided.

1. The 'Net Generation's' Challenges to Higher Education

Free / Libre Open and Source Software (FLOSS) communities succeed in providing and distributing in a sustainable manner the knowledge necessary for the production of good quality software, thereby using a different development approach than proprietary software producers (Demaziere 2006, Krogh 2003, Lakhani & von Hippel 2002). One characteristic of FLOSS is that it is built by a community of volunteers and frequently backed by companies that generate their revenues by providing services related to FLOSS projects and the software (Michlmayr,2004).

In more recent years FLOSS communities also gained attention for their community production and support models and regarding their way of knowledge creation, sharing, and learning opportunities (Ghosh & Glott 2005, Hippel 2002, Hemetsberger 2006, Hemetsberger 2004). Though FLOSS communities might be the most mature learning ecosystems; the web is not short to provide further success cases that show how knowledge can be created collaboratively by armies of volunteers, how user to user support systems work, and how sustainability and quality can be assured through community involvement (Barahona, Tebb & Dimitrova, 2005).

The coming 'net generation' is about to enter higher education. This generation not only grew up with ICT, but is also used to taking on an active role, to create and design resources, and to engage at the web with peers from all over the globe (Brown 2000). As Slot (2007) points out, this net generation is further used to two fundamentals of the so called web 2.0. Web 2.0 services are *open*, which means that in general there should be no access restrictions to participate at a given community; e.g. due to prior education, age, culture, or professional position; and they are predominantly *free*; both in terms of the freeness of accessing content and communities (free in monetary terms), but also freeness in terms of freedom to express oneself and to be creative.

So how will HE respond to those demands taking into consideration that the net generation is well aware about the options and alternatives the web provides?

Challenges to HE in this respect are:

Acting in the virtual world challenges traditional laws: The knowledge is power rule, for example, only applies if knowledge is being shared with others, but not by 'hoarding knowledge' as this means to remain invisible; with the knowledge being provided by someone else (Wayner 2000). As a consequence 'selling knowledge' is equally difficult in the virtual world as someone else might be willing to provide the same knowledge for free.

In times when knowledge is becoming obsolete faster and faster a 4 years' university student enrolled for a technical degree likely might face that half of what has been learned during the first year will be out of date by the third year of study.1 Educational settings will need to adapt new structures and models to keep the pace. Education at large struggles to update their courses within shorter and shorter cycles or to develop new ones, with lessons still being largely given like 100 years ago (Sowe & Stamelos, 2008a).

One certainly could argue that HE can not be benchmarked with informal learning environments the web provides. After all, HE is a full service provider and offers recognized degrees that can not be compared to self studying at the web. However, the case of computer science education research has shown that 39% of surveyed IT companies (Ghosh & Glott, 2005) expressed that there is no difference between formal qualification and practical experience in FLOSS, with a further 17% claiming that formal qualification is even worse. This indicates that HE is well advised to keep an eye on the learning opportunities the web provides, especially in contexts

[1] See http://www.slideshare.net/jbrenman/shift-happens-33834, slide 51.

where practical experience is considered equally or even more important than "theoretical" education at school or university.

Understanding web success cases like e.g. FLOSS is therefore crucial for HE to adapt itself to the new realities. The FLOSS case, but also the web at large, provide an insight on how HE might benefit from going a step beyond the current Open Educational Resource (OER) move towards Open Participatory Learning Ecosystems (OPLE) (Brown, 2007) and an educational commons (Hepburn, 2004).

2. FLOSS communities as open participatory learning ecosystems

Generally, FLOSS communities consist of individuals who contribute to, write, and build a particular application by means of the FLOSS development or bazaar model (Raymond, 1998). However, FLOSS is not only about software; and the capacity to integrate and play a role in the FLOSS community is not only dependent on good programming skills. Participating in FLOSS can also require expertise in patents law and license issues or management skills and capacities to mobilise the community as a social movement. Language skills are also required in the FLOSS community because many software projects ask for translations of the code and documentation into other languages ("localisation"). All these skills can be learnt within the community, through interaction with other community members and project participation (Ghosh & Glott, 2005; Glott, Meiszner & Sowe, 2007).

Learning is the most important driving force of the FLOSS community, as improving skills and sharing knowledge are by far the most important motivators for people to engage in FLOSS. FLOSS communities are thus virtual communities where the focus is on software development and related activities. Virtual communities and communities of practice (Brown & Duguid 1991) serve as important learning environments. Collaborative learning and the peer review process emphasize the importance of shared dialogue. The FLOSS community therefore serves as a model for the creation of self-learning and self-organizing communities (Sowe et al., 2004, 2005). In this regard, the principles and practices of learning in the FLOSS community appear helpful to master the challenges coming up with the growing demand for "lifelong learning", particularly the necessity of new leaning arrangements that are more informal, self-organized, and incidental (i.e. driven rather by situational personal interests and needs than by pre-defined curriculae of educational institutions or firms) (Keeton et al., 1976; Houle, 1976; Chickering, 1976; Coleman, 1976, 1995; Lave & Wenger, 1991; Watkins & Marsick, 1992). David & Foray (2002) describe this change as an overall shift from "learning to do" to "learning to learn".

Virtual communities, such as the FLOSS community, allow both, learners and instructors, to try new approaches (Glott, Meiszner & Sowe, 2007). Collaborative learning and the peer review process emphasize the importance of shared dialogue which results in the creation of public knowledge resources for the benefit of interested individuals. The FLOSS system is based upon the commons' component ; starting with the aspect of commons based peer production, the release of the final product to the commons and ending with commons support system. (Lakhani & von Hippel 2002; Hemetsberger 2006) 'Collaborative content creation', 're-use' and 'peer review' are key factors of the FLOSS production process and deeply embedded within the FLOSS community structures. Learning in the FLOSS community is characterised by

1. Openness and inclusiveness; everybody can join and contribute (Giuri et al. 2004)2
2. Up to date and dynamic content; everyone can add, edit and update the content
3. Materials are usually the product of many authors with many contributions from people other than authors
4. Frequent releases and updates based on a continuous re-negotiation/reflection process within a continuous development cycle
5. Prior learning outcomes and processes are systematically available through mailing lists, forums, commented code and further instructional materials (re-use)
6. A large support network; provided voluntarily by the community member in a collaborative manner nearly 24/7
7. Lurkers3 are welcome paradox – the more the better
8. New ICT solutions are adopted early

[2] This may not apply to source code repositories, where access can be restricted to core developers (though others may contribute through these core developers). However, access to mailing lists, forums, other project resources etc. is usually not restricted.

[3] In Internet culture, a lurker is a person who reads discussions on a message board, newsgroup, chatroom, file sharing or other interactive system, but rarely participates. See http://en.wikipedia.org/wiki/Lurker

FLOSS communities provide thus a good example for open participatory learning ecosystems (Brown 2007) in which users become active 'resource' creators, learning processes are made visible for other learners, and user support systems are established and maintained in a sustainable manner.[4] Learning in the FLOSS community highly corresponds to the definition of OER, which is "the open provision of educational resources, enabled by information and communication technologies, for consultation, use and adaptation by a community of users for non-commercial purposes"(Holotescu 2007).

Concluding, FLOSS challenges and influences teaching and learning as well as content quality and delivery at a rate unprecedented in the history of both software development and education. It challenges traditional education systems as these are *per se* closed systems where content is usually developed by the educational provider only and re-used or reviewed within a clear defined scenario. And it also challenges the OER movement, as this is by now still based on the traditional production model considering only educators and professionals.

3. The FLOSS software & content production model and its applicability in HE

FLOSS communities typically have tools at their disposal which are not only used for software production but also as mechanisms to coordinate massive amount of individual efforts. These tools include versioning systems (CVS or SVN), bug-tracking systems, mailing lists, forums, to-do lists, etc. In addition to enabling the software development to proceed, FLOSS communities also provide users and novices with various types of learning resources, like manuals, tutorials, or wikis (Weller & Meiszner, 2008). The explicit knowledge in these tools or repositories is constructed as a result of continuous socialization, discussion and negotiation between community members. That is, they are jointly generated by user and developer and after generation continuously updated and improved. As illustrated in Figure 1, project participants socialize by sharing their knowledge. Individuals make their tacit knowledge explicit to the project through externalization. Combination refers to the formation and organization of abstract knowledge from explicit knowledge. Through internalization

[4] http://sweng.csd.auth.gr/~sksowe/Publicat/IDBK001-C16_285-303_.pdf

individuals will absorb explicit knowledge, combining it with their own knowledge and experiences to produce "new" tacit knowledge. Through active participation, individuals' tacit knowledge is transformed into explicit knowledge.

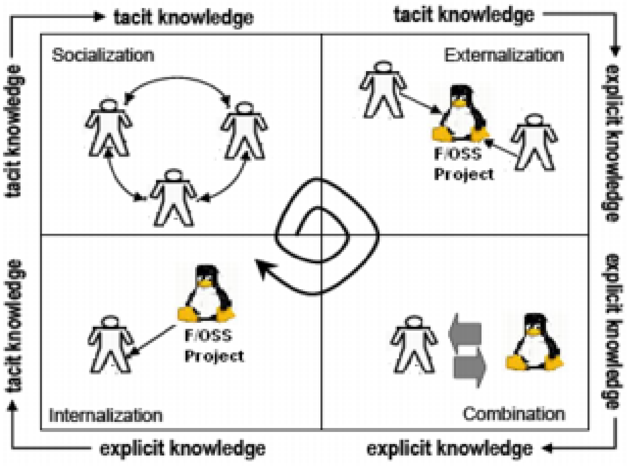

Fig. 1. Generating and using contents through interaction in FLOSS projects (Sowe & Stamelos, 2008b)

Compared to traditional static software production mode, Scacchi (2002) introduced "software informalisms" to describe the FLOSS mode of producing (software) product requirement definition, the sense making involved the production process, continuous discourse between all participating partners , and how FLOSS community members become accountable for the content they produce. As noted by Scacchi, the requirements for a FLOSS product are, unlike for traditional software products, not pre-defined, but specified through developer and user discourse: "email or bboard (forum) discussion threads, system vision statements, etc.

From the learning point of view, informalisms might help in understanding the type of "Learning Resources" that users in FLOSS in general dispose of. The software programme itself might be seen as an analogue to the content of a course in formal education. But unlike in education, or even in formal software development, there is no "Requirement Specification" document for FLOSS products. Instead users and developers are in a constant re-negotiation of the software's features, functions or design (Scacchi 2002). Following tools and content production methods in FLOSS might be applicable to HE:

Communications and Coordination Technologies: Mailing lists and forums are the common place for community communications to discus about the requirement of the software or known bugs, but also other organizational aspect and they are also the main place to provide support to users. Chats, instant messagings or voip are also used but for more ad-hoc discussions. The advantage of communications in mailing lists and forums is that other users can later on read through these.

Hemetsberger (2006) suggests that members of FLOSS communities learn and build collective knowledge through the use of ‘technologies’ and the establishment of discursive practices that enable virtual re-experience. Following the problem solving processes, or other type of argumentation lines, are important learning resources of FLOSS communities that enable other users’ re-experience. By these means, users get access to “knowledge that is often tacit in nature but visible and observable in the common practice of and interactions among competent practitioners”, which is “also highly contextual and, therefore, cannot be externalized and taught independently from its context” (Brown & Duguid 1991). Many community members are well aware of the role of mailing lists and forums and consequently expect that these resources are used first, before individual support might be provided.

Scenarios of usage as linked Web pages : To explain the functioning of the software “community participants create artifacts like screenshots, guided tours, or navigational click-through sequences (e.g., “back”, “next” Web page links) with supplementary narrative descriptions in attempting to convey their intent or understanding of how the system operates, or how it appears to a user when used...participants may publish operational program execution scripts or recipes for how to develop or extend designated types of open software artifacts” (Scacchi 2002). As a use case live demo versions are also commonly available where users can log in at the front and back-ends to experience the software in practice.

HowTo Guides: How to guides are also provided that explain how the software functions. Additionally communities might make use of FAQs, knowledge bases or wikis (Meiszner 2007). Further valuable “How To Guides” are also the community forums.

Open Software Web Sites and Source Webs: Community websites have the advantage to provide the community with an information infrastructure for "publishing and sharing open descriptions of software in the form of Web pages, Web links, and software artefact content indexes or directories. These pages, hypertext links, and directories are community information structures that serve as a kind of organizational memory and community information system. Such a memory and information system records, stores, and retrieves how open software systems and artefacts are being articulated, negotiated, employed, refined, and coordinated within a community of collaborating developer-users" and it might "include *content* that incorporates text, tables or presentation frames, diagrams, or navigational images (image maps) to describe their associated open software systems. This content may describe vision statements, assert system features, or otherwise characterize through a narrative, the functional and non-functional capabilities of an open software system…Web content that describes an open software system often comes with many embedded Web *links*. These links associate content across Web pages, sites, or applications"(Scacchi 2002)

4. The FLOSS support model and its applicability in HE

The FLOSS support model is already, through its conception of being provided on a voluntary base, very different from the, at least officially, guaranteed support system in educational settings. Personal support in FLOSS is in general provided through asynchronous communication tools such as mailing lists, forums or wikis; with the benefit of preserving the various answers and the process towards reaching the answers. Those recorded answers and documented processes are then becoming valuable learning resources for future learners, with individuals acting as knowledge brokers and point one the other to useful sources or leverage knowledge from one domain to another (Sowe et.al., 2006; Scacchi 2002, 2006)

Though synchronous (direct) personal support, as is typically provided in traditional educational settings, can be found too, but to a much lower degree. This support model provides risks as well as chances to learners. A risk is that learners must evaluate which source of information and support they can trust (there is no approval of teaching material by teachers or the like). Chances are a gain in information sources and the speed in which the community reacts to individual information demands, 24 hours / 7 days a week. The ubiquity of information resources and support is probably one of the strongest challenges for traditional HE.

As of today there seem to be few cases within the educational landscape that try to provide similar extra-institutional (FLOSS-like) community based support environments that are open to participants other than formally enrolled students. Two major support environments in this respect are the Utah State University's Open Learning Support (OLS)[5] and the OpenLearn initiative from the British Open University[6]. Despite the scope of this two initiatives the degree of usage of provided asynchronous communication tools is much below the degree of usage one can find at the myriad of informal support communities the web provides (like e.g. Jishka[7]).

Cases like FLOSS, or also Jishka, demonstrate that volunteering support environments can work out for education, meanwhile cases such as OLS and OpenLearn demonstrate the need for further research and piloting on how to establish equally vivid and functioning environments within the educational landscape.

5. The FLOSS business model and its applicability in HE

A fundamental distinction between FLOSS and proprietary software is that proprietary software is exclusively produced in firms, whereas FLOSS is produced by the FLOSS community (a diverse group of developers) and in firms. As Krishnamurthy (2003) points out, FLOSS community members develop software rather because they share a passion for the product than to make profit. Furthermore, they do not make a distinction between individual and corporate users, which is why the product as well as the source code is made freely available to any (kind of) interested user. Krishnamurthy describes the FLOSS community as indifferent to its own profits as well as to profits that corporations can make from its products. Instead of profits the community appears to be interested in widespread adoption of its products. The way how to keep control over the products is the license.

[5] http://ols.usu/edu/courses
[6] http://www.open.ac.uk/openlearn/home.php
[7] http://www.jiskha.com

According to the FLOSSIMPACT Report,[8] the market for Open Source Software has considerably accelerated, due to factors such as availability of high-quality software, low cost and low barrier to entry, availability of customisation and local support services, and vendor independence and flexibility. The report further shows that FLOSS has shifted from a model driven purely by the developer community and university support to one where a main driver is industry, including businesses typically devoted to development and support of specific products or to maintenance and integration activities as well as large firms, including such major industrial players as IBM, Oracle, Philips, Nokia and SAP. The report concludes that FLOSS has considerably reshaped the business models strategies of such large companies. The development model and licensing terms naturally provide preference to service-oriented business models where the core profit centre is not pure software development. Some smaller firms have also successfully followed a business model based on pure software sales through a process of dual licensing (GPL + proprietary) – the best known of these being MySQL.

The FLOSS User Survey[9] distinguished two fundamentally different groups of firms involved in FLOSS-related businesses: One group tends to have considerable product and technology knowledge, based on significant knowledge of and involvement in FLOSS, which is used to build up what is essentially a services business. These include firms that provide training, support, consultancy and integration, as well as extending hardware sales that are enhanced by their expertise in terms of technical knowledge of FLOSS and participation in the FLOSS development community. This group includes very big companies, such as IBM, as well as very small companies like Linuxcare. A subset of this group of firms works exclusively with niche FLOSS-only product development. Examples of such firms are MySQL, JBoss, ORIXO, and ZEA Partners.

The other group consists of firms with considerable expertise in services and integration that are broadening their service provision by adding FLOSS to their portfolio. Examples of such firms are KPMG consulting, Cap Gemini Ernst & Young, or smaller focussed firms such as Microconsult in e-learning or Monster.de in

[8] See http://ec.europa.eu/enterprise/ict/policy/doc/2006-11-20-flossimpact.pdf
[9] See http://www.berlecon.de/studien/downloads/200207FLOSS_Use.pdf

recruitment). This often allows firms to provide more complex services and integration possibilities while retaining a larger share of profits within their firm, as they essentially retain 100% of the profits and do not have to pay royalties to other (proprietary) firms for use of software.

With the transition to service based business models, the software itself becomes more and more a platform on which services can be based (Glott & Ghosh 2007).

The trend towards service-oriented business models opens some opportunities for HE. Services might be provided to commercial companies for customized training, for prescribed learner support plans, or for offering certification of informally acquired skills and resulting credit points to free learners outside of formal education. In the case of Linux for example DELL offer PCs with a pre-installed Linux Ubuntu distribution and allows customers to choose between the freely available volunteer support system or alternatively paid support subscription plans. Looking beyond the current OER approaches those mixed business models might be an additional model for HE, as the knowledge that is taught and learnt is no longer a means for the production of goods that can be sold but the knowledge itself becomes a good.

Furthermore, technologies and solutions used in FLOSS products tend to be characterized by a relative longevity, when compared with their proprietary equivalents, due to possibility of their adaptation to changing market or technical conditions. This has important consequences for education, as skill sets related to FLOSS technologies should be more useful for students (and subsequently more useful on job market) in the long term. This is obviously highly dependent on usage of FLOSS products in the industry (Glott & Ghosh 2007).

6. Conclusion

The background of this paper is provided by the changing landscape and efficacy of educational systems, as the information society demands ever faster generation and dissemination of knowledge and increasing requirements from learners and educators to comply with this. The OER movement has emerged as an alternative to traditional educational environments, aiming at opening the door to the next generation of HE provision.

However, the examination of FLOSS as an example of a well functioning open participatory learning ecosystem (OPLE) has revealed that by now the OER movement did not tap the full benefits provided by Web 2.0, which are already widely used by the 'net generation'. The FLOSS case provides us with insights in how to make use of ICT to provide students and free learners outside formal education with learning opportunities that are embedded in global virtual OPLE. Free, open, transparent, inclusive and sustainable are just five of the keywords that relate to those approaches and that might be taken forward to educational settings.

There are many advantages for learners in FLOSS communities, such as access to a variety of resources, learning by doing, community support, mentorship and engagement. Educators would benefit from access to a large pool of up to date learning materials / content and the community support system.

It has been shown in this paper that FLOSS learning principles and practices challenge traditional as well as modern (OER-related) educational approaches. But the FLOSS-type of learning is not radically new and unrelated to the solid pedagogic framework that has been established for new types of learning, as a response to the shortcomings of traditional educational ecosystems. FLOSS appears not as a contradiction to these pedagogies but in many respects as a best practice case of their principles and goals. However, there is still the need for further research and piloting to better understand the applicability of FLOSS-like approaches to formal educational settings, or to establish open participatory learning ecosystems that go beyond the current open educational resource movement and that are self-sustainable. The maybe most relevant characteristics of FLOSS communities that could help to improve (higher) education and to meet the net generation's expectations are likely the community production model, the community support model, and the underlying business models to assure sustainability.

Bibliography

- Brown, J. S. (2007) “Open Learning Broadly Construed”, Keynote at the OpenLearn conference 2007, 30-31 October 2007, The Open University, UK

- Brown, J.S. (2000) “Growing up digitally: How the web changes work, education and the ways people learn”, CHANGe, 32(2), 10-20.

- http://www.usdla.org/html/journal/FEB02_Issue/article01.html (Retrieved 27 Feb 2008)

- Brown, J.S. and Duguid, P. (1991) “Organizational Learning and Communities-of-Practice: Toward a Unified View of Working, Learning, and Innovation”, Organization Science, Special Issue: Organizational Learning: Papers in Honor of (and by) James G. March, 2 (1), pp.40-57.

- Chickering, A. W. (1976): “Developmental change as a major outcome”, Keeton, M. T. et al. (eds.): *Experiential learning: rationale, characteristics, and assessment*. San Francisco. 62-107

- Coleman, J. S. (1976) “Differences between experiential and classroom learning”, Keeton, M. T. et al. (eds.): *Experiential learning: rationale, characteristics, and assessment*. San Francisco. 49-61

- Coleman, J. S. (1995) “Experiential learning and information assimilation: Toward an appropriate mix”, Warren, K.; M. Sakofs & J. S. Hunt Jr. (eds.): *The theory of experiential learning*, Boulder, Colorado. 123-129

- David, Paul A. & D. Foray (2002) “Economic fundamentals of the knowledge society” Available at http://www-econ.stanford.edu/faculty/workp/swp02003.pdf

- Demaziere, D. (2006) "How free software developers work”, Môle Armoricain de Recherche sur la Société de l’Information et les Usages d’Internet

- Ghosh, R. & R. Glott (2005) “FLOSSPOLS Skill Survey Report”, www.flosspols.org (Retrieved 27 Feb 2008)

- Giuri, P.; M. Ploner; F. Rullani; S. Torrisi (2004) “Skills and Openness of OSS Projects: implications for performance”, LEM working paper 2004/19, Sant’Anna School of Advanced Studies, Pisa. Available online at http://opensource.mit.edu/papers/giuri_etal.pdf

- Glott, R. & R. Ghosh (2007) “FLOSS”, 3S Green Paper on Software and Service Architectures, Infrastructures, and Engineering, pp. 92-105. Available at http://www.eu-ecss.eu/contents/events/green-paper/WP1_D1.2_Green%-20Paper_v_ecss_websitepdf.pdf

- Glott, R.; A. Meiszner & S. K. Sowe (2007) „FLOSSCom Phase 1 Report: Analysis of the Informal Learning Environment of FLOSS Communities”, FLOSSCom Project. 2007. Available at http://opensource.mit.edu/papers/FLOSSCom_WP2_Phase_1_Report_-v070709_1.pdf (Retrieved 27 Feb 2008)

- González-Barahona J. M.; C. Tebb, V. Dimitrova, D. Chaparro and T. Romera (2005b) “Transferring Libre Software Development Practices to the Production of Educational Resources: the Edukalibre Project”, *Edukalibre project* http://edukalibre.org/documentation/edukalibre-4-page_system-2005.04.17.pdf (Retrieved 30 Nov 2007)

- González-Barahona J. M.; C. Tebb; V. Dimitrova, D. Chaparro, and R. Mazza (2005a) “Producing educational Resources in the “Libre Way””, *Edukalibre projec*”, http://edukalibre.org/documentation/iadis_slides.pdf (Retrieved 30 Nov 2007)

- Hemetsberger, A. (2004) “Sharing and Creating Knowledge in Open-Source Communities: The case of KDE”, http://opensource.mit.edu/papers/hemreinh.pdf (Retrieved 27 Feb 2008)

- Hemetsberger, A. (2006) “Understanding Consumers' collective action on the internet: A conceptualization and empirical investigation of the free and open-source movement” Research Synopsis, Cumulative Habilitation at the University of Innsbruck, April, 2006, http://www.hemetsberger.cc/publications/index.html (Retrieved 27 Feb 2008)

- Hepburn, G. (2004) “Seeking an educational commons: The promise of open source development models”, First Monday, Volume 9 (Number 8).

- Holotescu, C. (2007) “Open educational resources and FLOSS”, Paper presented at eLiberatica, Brasov, May 2007

- Houle, C. O. (1976): “Deep traditions of experiential learning”, Keeton, M. T. et al. (eds.): *Experiential learning: rationale, characteristics, and assessment*. San Francisco. 33.

- Keeton, M. T. et al. (eds.) (1976): *Experiential learning: rationale, characteristics, and assessment*. San Francisco.

- Krishnamurthy, S. (2003) “An Analysis of Open Source Business Models”, available online at: www.linpro.no/.../¬download/646/¬4790/version/¬1/file/¬Analysis%20¬of%-20¬Open¬%20Source%20Business%20¬Models.pdf

- Krogh, G. v., S. Spaeth, K. R. Lakhani (2003) “Community, joining, and specialization in open source software innovation: A case study”, MIT Sloan Working Paper No. 4413-03.

- Lakhani, K. R. & E. von Hippel, (2002) "How Open Source Software Works: "Free" User-to-User Assistance." Research Policy, 32, 923-943.

- Lave, J. and Wenger, E. (1991) *Situated learning - Legitimate peripheral participation,* Cambridge: University of Cambridge Press.

- Meiszner, A. (2007) “Communication Tools in Floss communities: A look at Floss communities at large - beyond the development team”, Web Based Community 2007 Conference, Salamanca - Spain.

- Michlmayr, M. (2004) “Managing Volunteer Activity in Free Software Projects Proceedings of the 2004 USENIX Annual Technical Conference, Freenix Track”, 93-102.

- Raymond, E. S. (2001) *The Cathedral & the Bazaar: Musings on Linux and Open Source by an Accidental Revolutionary*, San Francisco: O’Reilly

- Scacchi, W., (2002) “Understanding the Requirements for Developing Open Source Software Systems”, IEE Proceedings--Software, 149(1), 24-39, February 2002.

- Scacchi, W., (2006) “Understanding Free/Open Source Software Development Processes”, Software Process Improvement and Practice, 11 (2), pp.95-105.

- Slot, M. F., Valerie (2007). "Users In The 'Golden' Age Of The Information Society", OBS-Observatorio, Special Issue 'Users as Innovators' Vol. 1(No. 3): 201-224.

- Sowe, S. K. and I. Stamelos. (2008a) “Involving Software Engineering Students in Open Source Software Projects: Experiences from a Pilot Study”, *Journal of Information Systems Education* (JISE), Vol. 18 (4), pp:425-435.

- Sowe, S. K. and I. Stamelos. (2008b) “Reflection on Knowledge Sharing in F/OSS Projects”, Proceedings of the Fourth International Conference on Open Source Systems, 7-11 September, 2008, Milan, Italy.

- Sowe, S. K.; L. Angelis; I. Stamelos. (2006) “Identifying Knowledge Brokers that Yield Software Engineering Knowledge in OSS Projects”, Information and Software Technology, Vol. 48, pp 1025-1033.

- Sowe, S.K., A. Karoulis, I. Stamelos, and G. L. Bleris (2004) “Free/Open Source Software Learning - Community and Web-Based Technologies”, IEEE Learning Technology Newsletter, Vol. 6(1), 2004. pp 26-29.

- Watkins, K. E. & V. J. Marsick (1992) “Towards a theory of informal and incidental learning in organisations”, International Journal of Lifelong Education, 11/4, 287-300.

- Wayner, P. (2000) *Free for All - how Linux and the free software movement undercut the hightech titans*, 1st ed., New York, NY: HarperBusiness.

- Weller, M. & A. Meiszner (2008) “FLOSSCom Phase 2: Report on the effectiveness of a FLOSS-like learning community in formal educational settings”, FLOSSCom Project. 2008.

- http://flosscom.net/index.php?option=com_docman&task=doc_download-&gid=183&Itemid=116 (Retrieved 14 March 2008)