PREVENTING UNWANTED COMMUNICATION IN ICT-BASED EXAMS BY USING FREE SOFTWARE

Leister, Wolfgang, Norsk Regnesentral, Postboks 114 Blindern, 0314 Oslo, Norway, wolfgang.leister@nr.no

Fretland, Truls, Norsk Regnesentral, Postboks 114 Blindern, 0314 Oslo, Norway and KnowIT Stavanger, St. Olavs gate 20, 4308 Stavanger, Norway, truls.fretland@knowit.no

Solheim, Ivar, Norsk Regnesentral, Postboks 114 Blindern, 0314 Oslo, Norway, ivar.solheim@nr.no

Abstract

We describe an ICT-based exam system for the Norwegian school system that prevents candidates from using the diverse communication devices available on modern computers to cheat. Based on the requirements for such a system from the authorities we show various solutions. We developed the exam system by using free software and the peer production model. We discuss our experiences with free software for the public sector, and point out areas where free software still has challenges to overcome when introduced as an alternative.

Keywords: ICT-based Exam, Free Software, FOSS, Public Sector.

1 INTRODUCTION

The compulsory schools in Norway are migrating the national written paper-based exams to computers. Schools with ICT-based exams will only have Internet access for retrieval of exam questions and submission of exam papers. These rules apply nationwide for paper-based exams as well as exams performed on stationary computers or the candidates' own laptops (Norwegian Directorate of Education and Training 2009). However, the school authorities at the municipal level for primary or secondary schools, or the county level for high schools, are responsible to define how these exams are performed. The school authorities have defined different regimes to enforce the applicable rules; solutions span from logging the candidates' use of the computers during exams and closing access to the Internet during exams, to performing visual control by invigilators. These rules leave many loopholes for cheating in exams, and could result in identifying false positives, i.e., recognising cheating that did not take place voluntarily.

The technical solution that is presented and discussed here aims to eliminate these loopholes and provide a secure and efficient way of preventing unwanted communication in exams.

The research focus in the paper is on understanding and describing the complex and multidisciplinary process of developing, testing and implementing a free and open source software (FOSS) solution for preventing unwanted communication in ICT-based exams in Norwegian schools. ICT-based exams also raises a number of issues pertaining to education and pedagogy as well as challenges related to e-assessment and testing, but these questions go beyond the scope of this paper.

2 METHODOLOGICAL APPROACH

The methodological approach of the paper combines action research (see e.g , Reason, P. & Bradbury, H., (Ed.) 2001) and case study research (Yin, 1984). The case study research method has been defined as "an empirical inquiry that investigates a contemporary phenomenon within its real-life context; when the boundaries between phenomenon and context are not clearly evident; and in which multiple sources of evidence are used" (Yin, 1984, p. 23). Action research can be defined as "… a form of collective self-reflective enquiry undertaken by participants in social situations in order to improve the rationality and justice of their own social or educational practices, as well as their understanding of those practices and the situations in which the practices are carried out… The approach is only action research when it is collaborative, though it is important to realise that action research of the group is achieved through the critically examined action of individual group members (Kemmis and McTaggart 1988, p. 5-6). In the present work the case study approach has been supplemented with an action research approach. The case described in this paper is the process and context of developing, testing and implementation of an ICT artefact for preventing unwanted communication in exams in schools. Various factors, challenges and actors are described in detail and in their educational and organizational context. The paper is based upon results from a survey, carried out by the authors, about the role, scope and challenges of ICT-based exams in all nineteen counties in Norway. Responsible informants in the ICT departments in the counties were contacted and provided answers to central questions.

Importantly, the authors are themselves actors, not simply observers. The paper is based partly upon (two of) the authors' first hand experiences as software developers of the ICT-based exam system presented and discussed here. The authors have been actively involved in the process of local and regional adjustment and local implementation. Such a unique position provides valuable access to key information not only about specific questions related to the technical implementation, but to other types of challenges as well. Furthermore, the authors have also – as software developers – had the opportunity to observe the role of the central educational authority and the interplay between educational authorities at all levels: municipal, regional/county and state level.

The paper sheds light on a process that comprises various challenges in technical, organisational, knowledge management as well as security and privacy, in developing and implementing FOSS solutions in a traditional hierarchical public agency with a predominantly proprietary ICT infrastructure. Security requirements of performing ICT-based exams and comments on the possible solutions are also addressed.

3 ICT-BASED EXAMS IN NORWAY

The overall regulations on the use of aids in exams in schools are defined in *The Knowledge Promotion* reform which is the latest reform in the 10-year compulsory school and in upper secondary education and training. It introduces changes in substance, structure and organisation from the first grade in the 10-year compulsory school to the last grade in upper secondary education and training. Following the Knowledge Promotion reform, the Norwegian Directorate of Education and Training has decided that there will be two models concerning aids during written exams (Norwegian Directorate of Education and Training 2008): The first model allows all aids, except the Internet and other communication tools. The second model covers exams with two parts. In the first part only tools for writing are allowed whereas in the second part all aids are allowed, except the Internet and other communication tools. The Directorate decides which model that will be applied in exams in different subjects. In 2009 ICT-based exams are still optional.

In an ICT-based exam several IT-systems must work together: A central server at the education authorities is set up as an exam management system (EMS)[1] which performs access control, sends out the test items, and collects the answers from the candidates for further processing. This part, as well as preparing the exam text and giving grades, are beyond the scope of our paper since this functionality is provided by the Directorate for Education and Training. The Candidate-system gives the candidate access to the exam. The invigilators might use a surveillance system to detect cheating attempts, while the school authorities need a system to set up the exam.

The written exams in the compulsory schools in Norway are administered in classrooms, where each candidate takes the exam individually using a computer. In some of these exams the use of written aids on paper or files might be allowed, but the use of communication equipment, including the use of the Internet, is prohibited. In this setting invigilators are watching that no unintended aids are used by the candidates.

We looked at the available technologies to be used for the candidate systems in ICT-based exams: (1) automated software installation in a homogeneous environment using unattended install or similar technologies; (2) thin clients; (3) application streaming; (4) PCs running in kiosk-mode; and (5) live distributions. Using an infrastructure of stationary PCs in the school's infrastructure running in kiosk-mode would give the best level of control. However, this is both inflexible and costly since cost and maintenance for the infrastructure and premises must be considered. Therefore, we do not consider an infrastructure of stationary PCs as the main solution for performing exams in the compulsory schools.

Lately, stationary PC installations in schools are being replaced by laptops owned by the school authorities or by the candidates. This new pattern of PC ownership poses new challenges to performing ICT-based exams, since the school authorities have reduced control of what is actually installed on the candidates' laptop computers. In most cases, the infrastructure of the schools consists

---

[1] In Norway, this system is referred to as the PAS and PGS modules run by the Directorate of Education. For exams run locally at schools the learning management system (LMS) is used.

of a wireless network, and a few PCs for surveillance and administrative purposes. Before discussing the possible solutions we look into the functional requirements to ICT-based exam systems.

3.1 Requirements to ICT-based Exams

The overall requirements to ICT-based exams have been defined by the Norwegian Directorate of Education and Training see (Norwegian Directorate of Education and Training 2008), but there will be adjustments and variations in the local implementations. The exam system must (a) provide for the candidates a familiar working environment; (b) provide the possibility to deliver the result to the exam server of the directorate; (c) perform the exam without failing, and provide a backup facility; (d) prevent the use of prohibited communication aids; (e) give access to aids that are allowed; (f) prevent the use of prohibited applications; (g) provide a simple surveillance system for the invigilators; and (h) provide a simple system to prepare and set up exams.

Modern laptops are equipped with senders and receivers for wireless technologies such as IrDA (infrared), Bluetooth, IEEE 802.11 (wireless LAN, WLAN). For other technologies, like ICE, WiMAX and different 3G technologies, cards and dongles are available based on PCMCIA or USB. Using these communication facilities it is possible to set up communication to sites that are beyond the control of the school's network. Most school infrastructures use a WLAN to connect the PC to the school's network, thus the use of wireless networks cannot be banned in general.

The regulations require that the access to the Internet must be restricted during an exam. Only the access to the site where the results are delivered, and to allowed services like the access to dictionaries, is open. Access to community sites that allow personal communication between the participants, such as Facebook, Google or Wikipedia, is prohibited. While some schools already restrict the use of such sites in normal teaching situations, it is observed that students have set up mirror servers on their home computers to get around this restriction. Note that some learning management systems (LMS) allow communication between candidates which must be disabled in an exam situation.

In addition, access to sites that provide ready-to-use essays and other sites that support plagiarism (Roberts 2007) must be restricted. Methods to cope with plagiarism are beyond the scope of this paper.

Written exams are performed with invigilators and technical personnel on site. While this eases the security requirements we notice that the invigilators in general do not have as extensive knowledge of IT as some of the candidates might have. Therefore, the invigilators often cannot detect cheating by visual inspection. A more thorough inspection of a machine might give the candidate the disadvantage of an interruption of the exam, and give candidates with unusual but allowed hardware disadvantageous.

The exam system must allow the access to written material, i.e., files on the candidates' computers while the execution of disapproved program files on these computers or on external devices must be prohibited. Individual files by the candidates might be made accessible from the computers' hard drives, external memory devices, file servers or from the LMS. It is up to the school authorities to decide which of these methods to use.

3.2 Discussion of Solutions

There are several technical solutions available or implemented to address the requirements to the exam system. To get an overview of which solutions are in use in Norway, we interviewed the school authorities in the 19 counties of Norway. In all counties, one or more schools had carried out ICT-based exams, or will start do so in 2010. Most of the counties had technical methods to limit digital cheating, whereas a few relied on purely non-technical methods, such as regulations, contracts and trust. We restrict the discussion to the technical methods.

The school authorities in nearly all counties buy laptops that the candidates can lease during the three years of study in high school. This arrangement is subsidised by the government, and the candidates get the option to buy-out the laptops after graduating. Less than half of the counties allow students to bring their own laptop. It is worth noticing that if the laptop is owned by the candidate, this will limit the school authorities' rights to modify the content on the laptop; hence these schools have fewer solutions to choose from in order to restrict digital cheating. In counties where the authorities purchase the laptops the number of different laptop models used in the school is smaller compared to user-purchased laptops. This eases the configuration and administration of the laptops.

We encountered four different ways of limiting digital cheating on ICT-based exams: monitoring or restricting the network, and monitoring or restricting the laptops, in various combinations. By *monitoring* we mean that the candidate's actions are being watched, but not hindered in any way. By *restricting* we mean that the range of allowed actions that the candidates can perform is limited by technical means. The school network and the laptop can be subject to both monitoring and restriction. An overview of the different solutions and the number of counties that have implemented this solution is given in Table 1.

	School network	Modify laptop	Replace on laptop
Monitor	1	1	0
Restrict	14	5	1

Table 1: An overview of the security solutions for ICT-based exams

We give a brief overview of the security solutions in the counties, especially (i) *laptop modification*, (ii) monitoring the school network, (iii) restricting the school network, (iv) replacing the operating system on the laptop; and discuss their advantages and disadvantages. Stationary systems where the infrastructure is entirely under control of the school authorities are not considered here. Neither is the "solution" where the only restriction is to turn the schools network off, hence violating Requirement (b) in Section 3.1of electronic delivery of exam. Note that also Requirement (d) in Section 3.1 might be violated in that case, since the students can have access to other unapproved networks.

Five of the school authorities in the 19 counties control the laptop by installing a client that restricts and/or monitors the actions of the candidate.[2] We refer to this as *laptop modification*. This requires the school to be the owner of the laptop.

Monitoring networking traffic during the exam is another solution that is already used by some school authorities, also in addition to other solutions. Unwanted traffic will be logged and trigger an alarm so that the invigilators or technical personnel can check whether a cheating incident has taken place. A variant of real-time monitoring is to log all traffic during the exam, and only examine logs after the exam if there are concrete suspicions of digital cheating. However, it is a challenge to distinguish between cheating and involuntary network traffic from the candidates' PCs. For instance, PCs could be set up to connect to certain sites when starting up or at regular intervals in daily use, e.g., Facebook or the candidate's home computer. These incidents might trigger an alarm during the exam, causing disadvantages for the candidate. Other incidents might be caused by malicious software that might be on the laptops, like worms and viruses, to the detriment of the candidates. This scenario might, in the worst case, cause that the invigilators are overwhelmed in alerts so that the exam cannot be performed if several candidates suffer from such malicious software. In addition, the amounts of logs generated during an exam day can be huge, even for the technical personnel.

[2] The different clients that were mentioned were Netop School (http://www.netop.com/products/education/school.htm), LanSchool (http://www.lanschool.com), BrowseControl (http://www.browsecontrol.com). Other classroom management software found on the web includes SMART sync, Spiral universe, iTALC, NetSupport School.

Other monitoring solutions include the surveillance of the candidate's screens, either using (physical) mirror installations, or surveillance programmes that transfer the screen content to the invigilator. Often, multiple screen dumps are presented in miniaturised versions on the invigilator's screen. This solution requires the attention of the invigilators, including the knowledge how an allowed state on the screen looks like. According to the school authorities, the ICT competencies of the invigilators – often retired people – are quite low, and training them provides a challenge.

Data traffic outside the school's network is difficult to monitor. An exam candidate could set up additional routes from the candidate's laptop via Bluetooth or 3G-connections, or via wireless zones that someone might have installed in the vicinity. This is especially a problem in densely populated areas, where there are many open wireless networks. Since it is not viable to shield the exam room from all these networks, the school authorities could install a surveillance program on each laptop to monitor the TCP/IP stack, not unlike a virus scanner or firewall program. However, diversity of hardware, operating systems, software, and software versions require an infrastructure to provide updates. Notice that when data traffic is monitored on candidate-owned laptops the permission of the data inspectorate would be necessary.

Some of the counties in Norway only require that the schools restrict their network in the routers to approved addresses. While this is reasonable and necessary as an additional measure, this solution does not block for access to networks and communication beyond the school network. The solution is usually based on 802.1x, an IEEE standard for port-based Network Access Control (Wikipedia). The laptop is authenticated before it is given restricted access to the network through the use of a network access list on the schools router/switch. Some counties achieved additional security by removing the administration privileges for the candidate on the laptop, and pre-configuring the laptops to connect to the schools network. A few went even further, and purchased only laptops that lacked internal Bluetooth. Unfortunately, there are USB-dongles that provide Bluetooth, and if the drivers are preinstalled they might be used without administrator rights. Since the candidates have access to their laptop prior to the exam to add approved documents they also have access to altering its functionality, e.g., add software that allows cheating. Detecting such software, or altered setup files, would require programs like a malicious-software scanner. This results in conflicts of interest, since some functionality might be necessary for operation of the laptop in normal situations, but prohibited during an exam.

Given the disadvantages of the prior solutions we suggest to replace the installed operating system during the exam entirely with an operating system that is approved. After the exam the laptops must be in the same condition as before the exam. With this regime we avoid extended monitoring and modification of the laptop, since the operating system during exam is set up to contain exactly the software needed.

Replacing the operating system temporarily can be achieved by starting the laptop from (a) the network; (b) the CDROM drive; or (c) a USB device such as a memory stick. Note that this feature must be enabled in the BIOS of the laptop. Starting the laptop from the network as suggested in alternative (a) requires a cabled connection and a server machine that provides the operating system. Of the two other alternatives the USB device can be accessed for writing, which can be used for backup purposes during the exam.

From a technical perspective all operating systems could be delivered by these methods. However, the license policy of some vendors[3] might be an obstacle, and could cause extra costs for the school authorities. For operating systems based on free software, like GNU/Linux, extra licensing costs do not apply. Later, we will describe the development of a system that replaces the original software on a laptop during exams based on free software.

---

[3] According to the license agreements for Microsoft Windows™ each instance of the operating system running on a PC must be licensed. Unless there are other agreements this will cause extra licensing costs for the school owners for each exam system when using such a solution. The use of a license tailored for virtual machines could apply in an exam context, but requires that a licensed instance of the used Microsoft operating system is installed on this computer.

While the new operating system takes over the control of the laptop without altering the previously installed systems, it is required that this new operating system in fact runs directly on the hardware. Running the exam system in a virtual machine could give the candidates the opportunity to switch to other operating systems without being detected. Therefore, it is essential that invigilators and technical personnel ensure that the systems are started directly from the BIOS without a hypervisor being present.

In Table 2 we summarise the properties of the discussed solutions by evaluating how they address the given requirements.

Requirement / Solution	(i) modify	(ii) monitor network/laptop	(iii) restrict network	(iv) replace
(a) Familiar working environment	Yes	Yes	Yes	Possibly
(b) Deliver electronically	Yes	Yes	Yes	Yes
(c) Perform exam	Yes	Yes	Yes	Yes
(d) Prohibit communication	Partly	No	Partly	Yes
(e) Access to allowed aids	Yes	Yes	Yes	Possibly
(f) Restrict forbidden applications	Yes	No	No	Yes
(g) Surveillance system	Partly	Yes	Partly	Minimal
(h) Simple setup	No	Yes	Yes	Yes
Same application software as in daily work	Yes	Yes	Yes	Possibly
Detect digital cheating	No	Yes	No	Yes
Restrict digital cheating	Yes	No	Partly	Yes
Can utilise user-owned laptop	Not always	Yes/Not always	Yes	Yes

Table 2. Characterisation of solutions (i)–(iv) with requirements.

When asking our respondents whether they intended to use the software from our project, we recognised that they considered it as a finished product rather than a prototype using FOSS. Many of our respondents were not familiar with the working environment, even though it is possible to configure the look and feel of the operating system and its user interface. Some respondents claimed that it would be difficult to use. While this might be true for the prototype, it is clearly possible to make the prototype a lot easier to use after a community effort. Of the nineteen counties five were using OpenOffice.org, but for those that use Microsoft Office and other Windows dependent software this might be more challenging. However, many of these issues can be resolved using platform-independent software and emulators. The rest of the objections were mostly based on misconceptions, like, “we do not want to monitor the candidates”.

None of the counties inspect the PCs before the exam, nor set prohibited files into quarantine during the exam, and reconstruct these files after the exam. We refer to this process to as *“PC washing”*. In that case the candidates can use a familiar working environment during the exam, which is an advantage. However, while the principle sounds simple, the diversity of hardware and software, and the introduction of new technologies make the detection of unwanted content difficult. PC washing is personnel-intensive[4] , and might also result in computers that fail after this process, since system files might be altered involuntarily. Additionally, setting some files into quarantine, and reconstructing them after the exam might also result in data loss, or even make the system unusable. Since the laptops might be owned by others than the school authorities this might lead to legal problems, privacy problems, and to costly reconstruction of the previous state of the machines.[5] In our opinion the disadvantages outweigh the advantages.

[4] PC washing might also require much technical knowledge from the personnel that supervises this process. The software to perform this operation needs to be updated frequently, and thus be a costly alternative.

[5] As a scenario we may chose a candidate who uses a parent’s PC while her own PC might be under repair. Consider the case that an important file disappears due to the some technical problems during PC washing. In this case, the school authorities could be made responsible for the damages. Note also that personal data might be accessible and stored externally

4 FREE SOFTWARE FOR ICT-BASED EXAMS

According to Benkler (2002) the production of software is based upon in three different models; (a) managerial command systems like firms or organisations, where hierarchies define the line of command; (b) markets, where the concept of transaction costs define the production; and (c) peer production, where other incentives govern than in the other two models, and which is based on decentralised information gathering and exchange.

Representatives for systems being developed according to these models are easy to find. In the context of exam management systems, the development of the PAS and PGS modules[6] are typical examples of a managerial command system described in (a). These are developed according to the exact specifications of the directorate; markets do not apply in this context, since the number of instances used is limited in centrally controlled software that is supposed to be used nation-wide.

However, this centralised technical model collides with a decentralised institutional and legal model of educational organisations. Accordingly, in the Norwegian educational system each school authority is responsible for how to perform the ICT-based exams (Norwegian Directorate of Education and Training 2009). We observe that many smaller projects to develop exam systems are emerging, resulting in diverse, incompatible solutions. The absence of a central organising instance is the ideal basis for organising software development according to the peer production model (Bauwens, 2006).

It is not a primary task of the school authorities to develop and distribute software that they might need in the daily operation of teaching. However, developments done by teachers should be made available to colleagues and other schools, thus enriching the teaching by new developments. Examples of this are Delingsbazaren.no or the community around the mathematics teaching software GeoGebra[7]. Often, the incentive for the participants is not of a pecuniary nature, but to contribute to their environment; thus forming an important prerequisite for the peer production model.

Financial incentives for the school authorities to participate in the peer production of an exam system are reduced costs in the form of software licenses and reduced maintenance. As will be illustrated below, the software to be developed is more flexible and more easily customised to its environment.

5 EXPERIENCES FROM A PROTOTYPE

As much as possible, our prototype builds upon modules from other free software projects. We base our work on a customisation of Ubuntu[8] a distribution of Linux. Below, we show which components we used and customised in order to implement the exam system that now is under testing by selected schools. The exam system consists of (a) the systems for the candidates; (b) the monitoring system for invigilators; and (c) the system to produce images for booting from memory sticks.

5.1 Candidate System

The candidate system implements the user interface used by the candidates during the exam, containing all necessary software and access to allowed documents and the school network. It is

---

during PC washing, which poses a threat to the privacy. In this case also the permissions of the data inspectorate might be necessary.

6 PAS (Prøve Adminstrasjons System) and PGS (Prøve Gjennomførings System) are developed by the Directorate of Education to control the exams centrally, i.e., distribute the tasks to be performed in the exams, authenticating the candidates, and collecting the responses from the candidates.

7 www.geogebra.org

8 www.ubuntu.com

implemented as a complete operating system loaded from a memory stick during boot time. We chose Linux as a base for the candidate system due to its availability, openness and customisation possibilities. Since some schools might have chosen to use other operating systems this choice might conflict with the first requirement of providing a familiar working environment. However, we believe that the differences can be minimised by clever configuration of the desktop, by choosing the appropriate applications, and by the use of an emulator for platform-specific software.

It turned out that the main challenge for the schools that used the candidate system was coping with diverse hardware. In one of the test schools, the students have a very heterogeneous mixture of laptop computers. This makes configuration of the candidate systems challenging since a wide variety of hardware, and especially wireless network cards from several vendors must be supported. While tests have shown that more recent versions of Ubuntu have better support for different wireless network cards, some schools minimise this issue by leasing the same laptop type to all students.

Among the many different distributions of Linux we chose Ubuntu with the KDE as desktop environment, known as KUbuntu. Some of the main factors for our choice include variety of included applications, community support and configuration support. In addition, KDE offers extra configuration possibilities suitable for the exam scenario that we did not find with other desktop environments.

The customisation of the OS is done in order to meet the requirements for the exam system (as summarised in Table 2; and especially; (1) configure access to the school network; (2) remove access to all other network interfaces; (3) provide the needed applications for the exam; (4) remove prohibited and/or unused applications; (5) localise; use Norwegian language in both OS, applications and spell checkers; and (6) install software to send periodic status messages to the monitoring system.

5.2 Monitoring System

The monitoring system receives periodic status messages from the candidate systems. These messages are interpreted and displayed on a web page, one line for each candidate system. If a candidate system goes down, the monitoring system shows an error message and changes the colour of that client's line on the user interface.

5.3 Producing the System Images

The images to be put onto the memory sticks for the candidate systems are customised on a modified installation of Ubuntu. This system also provides a set of additional scripts and configuration files. It is based on the open-source software Ubuntu Customisation Kit (UCK).

The source file tree that is installed on the candidate's system consists mainly of three types of files: Files that are modified versions of the original Ubuntu system, additional scripts that are used by the candidate system, and configuration files to set up the wireless network. To support the peer production model we made this system available[9] publically.

5.4 Involving the Community and Testing

A FOSS project typically needs an environment, like creating a web site for distribution, bug handling, organising implementation of new ideas, and communicating with the community. Today, free services offer the necessary basic functionality for hosting projects following the peer production paradigm.

The key components were set up by a core team of two developers. The system was tested by a community of employees at several schools that are interested in the project. After a short introduction

[9] Project web page: http://code.google.com/p/digeks/

to the system architecture the participants in the community were able to customise the system for their own environment. This included such things as setting network keys, adding additional support for hardware, or setting up printing.

During the first test phase the exam system was introduced at two schools with teachers and school personnel participating. In the initial testing period, dedicated wireless networks were set up, and all available portable PCs of different brands were started to see whether the system could work. Tests showed that many PCs had problems booting from the memory stick, while others did not connect to the wireless network due to missing driver support in the operating system. In order to fix this our community partners contributed with adding the necessary drivers, as well as giving practical advice on the setup, or add additional services, e.g., access to printers.

In the second testing period, limited local exams with typically 30 candidates were performed, on a similar set of computers. This reduced the problem of hardware diversity. During this exam using two base stations, only minor problems occurred that could be fixed quickly.

A full-scale test with 125 candidates (143 machines were prepared), using six base stations, six different types of memory sticks on five different kinds of machines was performed in March 2009 without larger technical problems. Two PCs had to be replaced during the exam, likely due to hardware problems. The automatic backup onto the memory sticks ensured that most work was preserved from these machines.

Most of the comments from the community were about improvements of the invigilator system and how to present the state of the system during an exam with many candidates in several class rooms. This part of the exam system has potential as a separate development task in the future.

The candidates in the exam are indirectly participants in this project. While we cannot expect active contributions from this group, their experiences give valuable input for an evaluation of the system. We expected that candidates that did not use Linux before would have difficulties to adjust in an exam situation. However, this did not happen, since the look-and-feel of the user interface is kept minimalistic, and only uses common user interface elements to all systems. Also the use of OpenOffice for text editing did not cause any comments or problems from the candidates. This finding is in contrast to the guidelines of the Directorate of Education and Training, who claim that candidates have to use the system at the exam that they use in their everyday teaching situation.

6 DISCUSSION AND CONCLUSION

In this project, the benefits of free and open software as well as a commons-based peer production model was clearly demonstrated, but also several challenges of the peer production approach were illustrated. There are also managerial issues and decisions that must be made, and where a peer model is not sufficient.

It has been shown how the software developed in the project was based upon the general functional requirements from the Directorate of Education and Training described previously in this paper. In the past, Norwegian schools have been owned and run by the central government, but this was changed seven years ago when the ownership was transferred to the municipalities for the lower levels and the counties for the upper secondary level. As we have seen, an important premise underpinning the requirements is that the school authorities are fully responsible for the actual technical and organisational implementation of the exam system. The technical solutions must be developed and implemented locally, on municipality and county level. In this case, the peer production model was followed, once a first prototype was up and running. Before that point the development followed more or less a managerial model using public funding.

In the development of the free software described in this paper, the commons-based peer production model (Benkler 2002) was followed. In line with Benkler (2002, 2006) we may argue, also based on the experiences from this project, that the peer production has several important benefits compared to market and managerial models. A key difference between the three models is related to the actors' individual motivation. Since the actors involved are driven by other motivating signals (personal interest, social commitment. altruism etc) than those derived from market prices and managerial commands or directives, there will be an intrinsic interest and personal involvement that may make the peer model more efficient and with less transactions costs.

This affects the software production in several ways. First, it can be argued that the peer production model makes knowledge management in many software development projects more efficient, i.e., in relation to how the knowledge demands, available competencies and persons are matched. The peer model was in this case crucial because it provided and integrated relevant competencies in a way that focuses on actual, immediate problem solving and relevant testing rather than on detailed implementation according to a set of pre-defined requirements and deliverables. These matters are hard to measure, but it seems likely that a managerial as well as a market model would have been less efficient in time and money because they lack mechanisms for an efficient and seamless match between knowledge demands, available competencies and persons.

Second, the model promotes collaboration and exchange of knowledge between peers; contributing to construction of *communities of practice* (Wenger 1998) that comprise various actors involved in the development processes. This ensured that the results of the work carried out in the project will be maintained and transferred to new projects and development processes. A case in point is how competence and skills were organised and distributed in the present project. The relevant profile and competence level of the various active actors in schools and municipalities involved were not defined in advance by a central body outside the project. Instead they were defined and recruited throughout and as part of the process in creating and developing a community of practice for the project where both developers, users and managers in the municipalities and at various schools participated in developing, testing and customising the software. Successful customisation of the software to the actual needs and status of the local infrastructures in the various schools and municipalities requires a development and testing model that allows, promotes and encourages social and organisational learning and knowledge management. In this case, the peer production model provided a collaborative and constructive community of practice that ensured that the experiences and considerations made on the local level were gathered and finally implemented in the basic software modules or as integrated parts of the local infrastructures at various schools. The result was that the software functioned well according to both central requirements and local conditions and infrastructures. It is not likely that these results would have been achieved in a similar cost-effective manner if a managerial development model had been chosen.

The peer production model also provides challenges. Peer production is about the development process but will not in itself provide further growth and dissemination. In order to gain momentum and ensure sustainable growth, the peer production model must be supplemented with other measures and resources, such as marketing, political and organisational work, media attention, etc. Experiences from the present project clearly signal that the peer production model may collide with established software policies and licensing regimes in the educational sector. Politicians may express their support to free software in the public sector, but the processes to introduce such software are not yet in place. According a blogger with relevant experience and positions in the educational system (Hauge 2009) the approach that the present project represents is considered to be a challenge to the Microsoft dominance in this sector. In particular, using open source software in exams challenges the present licensing regime in the school sector which in effect prevents the schools from choosing alternatives to Microsoft Office products.

Finally, there are also several policy decisions related to functionality, security and privacy that must be made on a central level by the project owner and the central development team. In these cases a managerial model is relevant. While designing the exam system one of the goals was to keep monitoring of network traffic and actions to check the contents of laptops to a minimum, mainly inspired by two reasons. (a) The monitoring and checking actions require extended knowledge within data forensics on the side of the invigilators in order to be able to decide whether an incident has been a cheating attempt. In the national exams in Norway the invigilators are often retired people whom we cannot expect to have an extended knowledge within IT or data forensics. In modern computers many processes that trigger network traffic might be activated regularly without the knowledge of the candidates, in addition to the possible involvement of malicious software. (b) Monitoring and checking content of computers owned by the candidates requires the permission of the Data Inspectorate in order to guarantee the privacy of the data. By only checking the up-time of the laptops regularly no sensitive data from the candidates' laptops are transmitted. In general, the main idea was to avoid monitoring as much as possible by only offering the necessary functionality to perform the exam. If an incident should happen inspite of these measures it must be considered a serious cheating attempt rather than network traffic by coincidence.

Acknowledgements

The development of the ICT-based exam system is supported in part by the Fornyings- og Administrasjonsdepartementet of the Norwegian government, and financed by the Møre og Romsdal County. The authors express their thanks to Martin Hauge (Møre og Romsdal County administration), Asle Klock (Molde videregående skole), Vegard Fremstad (Kristen Videregående skole Trøndelag), Arne-Kristian Groven, and Trenton Schulz for their support. We also thank the participants in the poll among the informants at the Norwegian county administrations.

References

Bauwens, M. (2006) "The political economy of peer production". *Post-autistic economics review*, issue 37, pp. 33–44.

Benkler, Y. (2002) "Coase's Penguin, or, Linux and The Nature of the Firm". *Yale Law Journal*, issue 112, pp. 369–446.

Benkler, Y. (2005) *Common Wisdom: Peer Production of Educational Materials*. COSL Press, Logan, UT, United States.

Hauge, M. (2009) "Utdanningsdirektoratet, digital eksamen og ulike agendaer". http://ksikt-forum.no/artikler/2009/1/utdanningsdirektoratet,_digital_eksamen_og_ulike_agendaer, in Norwegian, last changed January 27, 2009, last accessed April 8, 2009.

Kemmis, S. and R. McTaggart (1988) *The Action Research Planner*, 3rd ed, Geelong: Deakin University.

Norwegian Directorate of Education and Training (2008) "Hjelpemidler til eksamen i Kunnskapsløftet". http://www.udir.no/upload/Eksamen/Hjelpemidler_til_eksamen_i_kunnskapsloftet.pdf, in Norwegian, September 2, 2009, last accessed June 20, 2009.

Norwegian Directorate of Education and Training (2009) "Bruk av datamaskin til eksamen i Kunnskapsløftet". http://www.utdanningsdirektoratet.no/Artikler/_Eksamen/Bruk-av-datamaskin-til-eksamen-i-Kunnskapsloftet/, in Norwegian, last changed February 2, 2009, last accessed April 8, 2009.

Reason, P. & Bradbury, H. (Ed.), The SAGE Handbook of Action Research. Participative Inquiry and Practice. 1st Edition. London: Sage, 2001.

Roberts, T. (2008), ed., *Student Plagiarism in an Online World: Problems and Solution*. Idea Group Inc (IGI), Hershey, PA, United States.

Wenger, E. (1998) *Communities of Practice: Learning, Meaning and Identity*. Cambridge University Press, Cambridge, United States.

Yin, R. K. (1984) *Case study research: Design and methods.* Sage, Newbury Park, CA, United States.