Debian Edu / Skolelinux ITIL Manual

Publish date: 2026-11-01

Contents

1 Knowledge sharing for centralised administration **1**

2 License **1**

2.1 Thanks . . . 1

2.2 Background . . . 2

3 Service support **2**

3.1 Service Desk . . . 2

3.1.1 Tasks and roles . . . 3

3.1.2 Expected time usage . . . 5

3.1.3 Check list . . . 6

3.2 Incident Management . . . 6

3.2.1 Check list . . . 7

3.2.2 Planning and implementation . . . 7

3.2.3 Activities related to operational events . . . 8

3.2.4 Roles . . . 8

3.2.5 Key points . . . 8

3.2.6 Tools . . . 9

3.3 Problem Management . . . 9

3.3.1 Procedures for problem management . . . 9

3.4 Configuration Management . . . 10

3.4.1 Planning . . . 10

3.4.2 Management of Configuration Items (CI) . . . 10

3.4.3 Planning and installation . . . 10

3.4.4 Check list . . . 11

3.4.5 Relations to other processes . . . 11

3.4.6 Tools for configuration management . . . 11

3.5 Change Management . . . 11

3.5.1 Activities . . . 12

3.6 Release Management . . . 13

3.6.1 Basic . . . 13

3.6.2 Definitive Software Library (DSL) . . . 13

3.6.3 Database for configurations and hardware . . . 14

3.6.4 Build management . . . 14

3.6.5 Testing . . . 14

3.6.6 Fall-back solution . . . 14

3.6.7 Advantages and possible problems . . . 15

3.6.8 Planning and implementation . . . 15
3.6.9 Activities . . . 16
3.6.10 Tools . . . 16
3.6.11 Relations to other processes . . . 16
3.7 Tools for operational support . . . 16
3.7.1 Type of tool . . . 17
3.7.2 Evaluation criteria when selecting tools . . . 18
3.7.3 Product training . . . 18
3.8 Planning at the start of the implementation of service support . . . 19
3.8.1 Implementing service support . . . 19
3.8.2 Feasibility study . . . 19
3.8.3 Determine current situation . . . 19
3.8.4 General guidelines for project planning . . . 19
3.8.5 Project review and reporting . . . 19

4 Service Delivery **19**
4.1 Service level management . . . 20
4.1.1 General checklist . . . 20
4.1.2 Planning . . . 20
4.1.3 Implementation . . . 20
4.1.4 The operational situation . . . 21
4.1.5 Content of the Service Level Agreement (SLA) . . . 21
4.2 Financial Management . . . 22
4.2.1 Budgeting . . . 23
4.2.2 Accounting . . . 23
4.2.3 Planning the accounting and billing . . . 23
4.2.4 Implementation . . . 23
4.2.5 Daily operation . . . 23
4.3 Capacity Management . . . 23
4.3.1 Monitoring . . . 24
4.3.2 Analysis . . . 24
4.3.3 Configuration . . . 25
4.3.4 Implementation . . . 25
4.3.5 Preparing the capacity plan . . . 25
4.4 Availability management . . . 25
4.4.1 Availability measurements . . . 26
4.4.2 Infrastructure . . . 26
4.4.3 «Single points of failure» . . . 27
4.4.4 Risk management . . . 27

4.4.5 Testing . . . 27
4.4.6 Design improvements . . . 28
4.4.7 Planning for availability . . . 28
4.4.8 Planning for recovery . . . 28
4.5 Service Continuity . . . 28

5 ICT Infrastructure Management 28
5.1 Design and planning . . . 29
5.2 Deployment . . . 31
5.2.1 Roles during roll-out . . . 31
5.3 Operations . . . 32
5.4 Configuration item . . . 32
5.5 Technical support . . . 33

6 A design and planning example 33
6.1 Background for the plan . . . 34
6.2 What's expected of the ICT tools and services . . . 34
6.3 Skills needs . . . 35
6.4 Investments . . . 36
6.4.1 Pupils . . . 37
6.4.2 Teachers . . . 38
6.4.3 Recommended technical development budget . . . 40
6.4.4 Software, learning platforms, and services . . . 40
6.4.5 Check list centralisation . . . 40
6.4.6 Software . . . 40
6.4.7 Learning platforms . . . 41
6.4.8 Online services . . . 41
6.5 Use of resources in operations . . . 42
6.5.1 Operations' roles . . . 43
6.5.2 Operation and support costs . . . 44
6.6 Summary of the options . . . 46
6.7 Recommendation . . . 46
6.8 Attachment . . . 46

7 Setting up infrastructure 46
7.1 Network architecture . . . 46
7.1.1 Solution . . . 47
7.1.2 Exception handling . . . 47
7.1.3 Verification . . . 47
7.1.4 Update the configuration database . . . 47

7.2 Server profiles . . . 47
7.2.1 Combi-server as a combined resolution . . . 47
7.2.2 Description of the profiles in Skolelinux/Debian-Edu . . . 47
7.2.3 Solution . . . 48
7.2.4 Exception handling . . . 48
7.2.5 Verification . . . 48
7.2.6 Update the configuration database . . . 48
7.3 Hardware servers . . . 48
7.3.1 Solution . . . 48
7.3.2 Exception handling . . . 48
7.3.3 Verification . . . 48
7.3.4 Update the configuration database . . . 48
7.4 Client computers . . . 48
7.4.1 Table of client types . . . 50
7.4.2 Solution . . . 52
7.4.3 Exception handling . . . 52
7.4.4 Verification . . . 52
7.4.5 Update the configuration database . . . 52
7.5 Switches . . . 52
7.5.1 Solution . . . 52
7.5.2 Exception handling . . . 52
7.5.3 Verification . . . 52
7.5.4 Update the configuration database . . . 52
7.6 Wireless access points . . . 52
7.6.1 Solution . . . 52
7.6.2 Exception handling . . . 52
7.6.3 Verification . . . 52
7.6.4 Update the configuration database . . . 52
7.7 Firewall(s) . . . 52
7.7.1 Solution . . . 52
7.7.2 Exception handling . . . 52
7.7.3 Verification . . . 52
7.7.4 Update the configuration database . . . 52
7.8 Routers . . . 52
7.8.1 Solution . . . 53
7.8.2 Exception handling . . . 53
7.8.3 Verification . . . 53
7.8.4 Update the configuration database . . . 53
7.9 Setting up a simple firewall . . . 53

7.9.1 Solution . . . 53
7.9.2 Exception handling . . . 53
7.9.3 Verification . . . 53
7.9.4 Update the configuration database . . . 53
7.10 Setup: . . . 53
7.10.1 Solution . . . 53
7.10.2 Exception handling . . . 53
7.10.3 Verification . . . 53
7.10.4 Update the configuration database . . . 53

8 Useful commands **53**

8.1 Support for 4 GB memory <-- included in configuration management . . . 53
8.1.1 Exception handling . . . 55
8.1.2 Verification . . . 55
8.1.3 Update the configuration database . . . 56
8.2 Administrating packages (apt-get) . . . 56
8.2.1 Exception handling . . . 57
8.3 Update the package archive . . . 57
8.3.1 Exception handling . . . 58
8.3.2 Verification . . . 58
8.4 Update to new packages . . . 58
8.4.1 Warning . . . 59
8.4.2 Exception handling . . . 59
8.4.3 Verification . . . 60
8.5 Summary of installed packages . . . 60
8.6 Find the name of a particular package . . . 60
8.7 Show available information about packages . . . 60
8.8 Installation of packages . . . 61
8.9 Removal of installed packages . . . 62
8.10 Install a specific package version . . . 62
8.11 Install a package using dpkg . . . 63
8.12 Search through files in a package . . . 65
8.13 Find which package a file came from . . . 65
8.14 Unpackaging files from a package without installing the package . . . 66
8.15 Make your own package mirror . . . 66
8.16 Secure login to the firewall (ssh) . . . 66
8.16.1 Exception handling . . . 67
8.16.2 Verification . . . 67
8.16.3 Update the configuration database . . . 67

8.17 Status summary for the firewall (Coyote) . . . 67
8.18 Next . . . 70
8.18.1 Exception handling . . . 70
8.18.2 Verification . . . 70
8.18.3 Update the configuration database . . . 70
8.19 Last . . . 70
8.19.1 Exception handling . . . 70
8.19.2 Verification . . . 70
8.19.3 Update the configuration database . . . 70

9 Copyright and authors **70**

10 Appendix A - Contract on operating Debian Edu / Skolelinux **71**
10.1 CONTRACT ON OPERATING DEBIAN EDU / SKOLELINUX . . . 71
10.1.1 Appendix 1 - Definitions . . . 71
10.1.2 Appendix 2 - Customer Obligations . . . 72
10.1.3 Appendix 3 - The Vendor's obligations . . . 72
10.1.4 Appendix 4 - Prices and terms of payment . . . 74
10.1.5 Appendix 5 - General provisions . . . 75
10.1.6 Appendix 6 - Contacts and addresses . . . 77

1 Knowledge sharing for centralised administration

Small organisations are in practice dependent on individuals and therefore vulnerable if someone leaves. A thorough and quality assured system administration handbook is therefore essential to ensure stability and continuity in operating practices. The Program for Digital Literacy has the primary objective of developing a set of recommended operating solutions, with guidelines that provide schools and educational institutions with stability and predictability so that computers, networks and basic services function properly.

The ITIL book contains guidelines based on "best practices", adapted for municipalities using free software like Skolelinux to run centralized networks across multiple schools. These guidelines are adapted for municipal and regional administrative centres. Many municipalities have only one part-time position for ICT operations in schools. In Norway there are more than 300 small and medium sized municipalities; usually they have 1-4 persons working full time with ICT in each municipality. Therefore, sharing expertise and experience between operating organisations is essential for all.

2 License

This document is written under the GNU General Public License version 3. It means that you have:

- The freedom to use the documentation for any purpose (Freedom 0).
- The freedom to study documentation and adapt it to your needs (Freedom 1).
- The freedom to forward copies so you can help your neighbour (Freedom 2).
- The freedom to improve the document and distribute it with your improvements to the public, so that the whole community benefits (Freedom 3).

These freedoms are explained on Wikipedia. Torgeir Kielland at the University of Bergen's Faculty of Law has analysed the GNU license or terms of Copyleft. He states that the GNU license is copyright relevant. In short, you can use everything in this documentation as appropriate. You must ensure that your contributions also receive a General Public License.

2.1 Thanks

Many have contributed to the documentation. Essentially it was written by Knut Yrvin and Andreas Johansen with many contributions from Klaus Ade Johnstad. Halvor Dahl, Skolelinux Drift AS is on the committee, and has made several contributions to structure and form, and content. In addition, there are contributions Snorre Løvås, UNINETT ABC, Finn-Arne Johansen from BzzWare AS, Ragnar Wissløff from LinuxLabs AS, and the reference group who participated in writing the documentation. The following participated in the reference group:

- Monica Larssen - Harstad municipality
- Aksel Celasun - Hurum municipality
- Trond Mæhlum - Kongsvinger municipality
- Bjarne Nielsen - Nittedal municipality
- Stein Lier - Akershus county

This documentation is maintained in a wiki. This is to ensure that operations staff can easily search for solutions to problems, update configurations, and so on.

See the copyright page for the copyright status of this document.

http://www.gnu.org/copyleft/gpl.html

2.2 Background

The Program for Digital Competence is the Norwegian Ministry of Education's ICT plan from 2004 to 2008. One objective is to develop a set of recommended operating solutions and appropriate guidelines. It will provide schools and educational facilities with stability and predictability, so that computers, networks and basic services function properly. Operating solutions must be adapted to the institutions' size and needs.

This documentation contains guidelines based on practices customised for ICT-services within municipalities and counties. It is also applicable for commercial operators. Many municipalities have only one part-time position for computer networking operations within the schools. Overall just 13% of the municipalities in Norway have more than 20,000 inhabitants; 73 % have less than 10,000. Usually 1-4 people work full time in ICT within the municipal administration. When it comes to schools, there is often only one part-time ICT position, which can cover approximately 500-800 client computers at 5-10 schools, with around 1700-3200 students and teachers using the system.

The documentation is also suitable for larger organisations. It is based on the ISO 20000 standard for ICT operations, also known as the ICT Infrastructure Library. See Wikipedia for more information about the standard itself: http://en.wikipedia.org/wiki/ITIL

The first edition of this document was finished 19th July 2006.

The document is maintained in a wiki and can be updated at https://wiki.debian.org/!DebianEdu/Documentation/nb/ITIL. The previous version is available from http://developer.skolelinux.no/itil/oldindex.html

The document will be translated into English using https://www.transifex.com/projects/p/itil-revitalization/ from March 2015.

3 Service support

As mentioned in the introduction, it is recommended to begin by establishing an office for centralized operations to allow you to manage tickets. The benefits of this come quickly and are visible, which is important for customer and user satisfaction.

Once the office is up and running with a sensible workflow for tickets (user requests and troubleshooting) you will move on to the biggest challenge for the organisation. As a rule, this is either change management or problem solving. Organisations with "cowboy" system administrators who come up with smart ideas and implement them without much testing, often begin with change management. For organisations suffering recurring outages, problem solving comes first.

Whatever you choose to start with, a certain amount of configuration management will be necessary. Managing configuration is critical to delivering the software and services for the user. Software must work as expected. In order to make beneficial changes, one must know the configuration of the different programs.

To manage configuration changes you may use a database (Configuration Management Data Base (CMDB)). Few people use a database for all the configurations, and neither do you have to add all configurations to one single database. It's fine to place configurations into multiple smaller and partly independent repositories. Some people, for example, store configurations and setups in version control. But even if you have different repositories, you may get greater benefits if you connect information from the different processes.

For users of Debian Edu, most service configurations lie within a specific directory (/etc). These may benefit from being collected and stored in a central version controlled directory. This makes it easier to restore lost services and setup machines if they are reinstalled. This applies both to servers and user laptops or workstations. As part of the backup system in Debian Edu, a backup is made of the setup directory /etc. But the backup system is nothing more than a database or a version controlled directory for configurations.

3.1 Service Desk

The Service Desk is where users ask questions or report problems. At school, the ICT-contact often forwards operational reports to the Service Desk. There may also be requests like setting up a new PC, or installing a program.

At school the ICT contact is the link to the Service Desk. The ICT contact also responds to the most common questions. Some questions are too difficult to manage at each school and must be forwarded to the Service Desk. It is important to

have good cooperation between the school ICT contact and Service Desk operators. Tasks that are too extensive or too difficult to solve locally should be passed to the Service Desk.

Users may also get direct answers from an operator at the Service Desk. All operational enquiries go to the Service Desk. Enquiries will be assigned a case number. Anyone who has registered a case will receive an e-mail confirming that the inquiry has been received. During consideration of the case, those working with it at the Service Desk may send status updates to the user.

This way, users get one point of contact, and service desk operators get an overview of all of issues. Operations can be expected to troubleshoot across all parts of the organisation. Periodically the team leader needs to go through all issues and solutions in order to prioritize debugging and to prevent re-occurrence of errors, in order to provide schools with a stable operating environment.

Incidents can be reported by phone, fax, email or web form. Incidents that are more urgent must be prioritized. Incidents that need to be resolved quickly are usually reported by telephone. Less important events are usually reported via e.g. email. A member of the support staff should be assigned to the incident and will need to ask the user questions to investigate the problem.

- Remember to be an active listener, not a passive one.

All enquiries should be logged, and an email confirmation should be sent. It is important that the user should feel safe, and information about what might be the problem should be communicated to them. When the enquiry arrives at the service desk, a brief description of the incident should be logged. The enquiry may be from the ICT contact at the school, or from someone with an agreement to use the service desk. The event logging should happen as soon as possible, and it should be assigned a case number. The user should get a confirmation by email copy that the matter has been received and assigned an appropriate case number.

Previously, enquiries were written in paper logbooks. Today software is used to record the enquiries in a "Request Tracker". It is crucial for operations to log enquiries. This is basically for error handling, user requests, and prioritization of the various incidents. Log entries are important to prevent recurring errors. Because operational events are periodically reviewed, an assessment of fixes and priorities can be made. The log also provides a basis for improving the service by debugging problem services and applications based on what users perceive as problematic.

Thus the log of requests is a basic and necessary tool for both users and the service desk. There are several freely available systems for logging requests with good documentation <<FootNote(RT Essentials: http://www.oreilly.com/catalog/rtessentials/chapter/index.html)>>. Skolelinux Drift uses RT <<FootNote(RT: Request Tracker: http://www.bestpractical.com/)>> to handle requests.

One important thing when starting up support is not to get too tough a start. Do not try to achieve everything at once; bet rather on "quick wins" that keep the user informed, and aim for quick response times. It is also important to clarify who the service desk should forward events to, if they can not solve the issue themselves. The support desk must also check whether there will be disruptions for the user. This makes it quick and easy to give feedback.

For users it is important that incidents are dealt with. For the service office it is important that the incidents are handled correctly according to the service level agreement, and that work requested beyond what was agreed to is handled between management at the school and the system administration organisation.

3.1.1 Tasks and roles

We recommend getting agreement on what duties the school's ICT contact has and which responsibilities those who work at the Service Desk shall look after. Schools often have little resources compared to what is common in municipal administrations or private companies. At the same time, schools usually have many more users and often more client machines than are in use in the rest of the municipality.

Tasks are distributed according to roles. By having clearly established roles it is easier to distribute tasks and ascertain the working capacity necessary to resolve operational tasks. Operational experience in municipalities and professional organisations shows that the following roles are common.

- ICT contact at each school. This is often a teacher with ICT-teaching and/or technical background.

- Operator(s) working in the central IT service. This is a person skilled in operations.
- ICT coordinator who organises the educational use of IT, and contributes towards plans for developmental, operational and educational use. Often this is a teacher.
- ICT responsible. This is usually the principal who is responsible for IT operations.

There follows an overview of the various everyday tasks, some of which are contracted out by municipalities.

ICT contact(s) tasks at each school:

- Oversee the school's server room.
- Be the school's contact at the municipality - report errors and outages.
- Perform simple maintenance tasks such as replacing mice and keyboards, upgrading thin clients, and simple patching.
- Be the school's superuser - able to advise colleagues about: user interfaces, e-mail, video projectors and relevant applications.
- Participate in ICT gatherings.
- Create and administer local users.
- Perform simple maintenance of printers.
- Create and manage email accounts.
- Perform simple commands and operations under guidance of an ICT-tutor.
- Facilitate the use of ICT in teaching.

The operator's tasks:

- Receiving incident reports and service requests.
- Mentor ICT contacts by telephone and e-mail.
- By arrangement, visit schools to troubleshoot defects and malfunctions on computers, printers and servers.
- Backups.
- Security software updates on the school's computers (servers and clients).

ICT coordinator's duties:

- Assist school management and ICT contacts in developing technical and teaching plans for ICT.
- Provide the service desk and management with guidance when selecting software and similar.
- Ensure that schools have appropriate ICT tools for teaching, and that computers and networks are appropriate for school subjects.
- Provide advice and guidance to operational services on the technical and pedagogical requirements for ICT in schools.

ICT-responsible duties (principal, headmaster, head of operational services):

- Make joint purchases of computer equipment and enter into joint agreements etc.
- Develop competence plans.
- Provide schools with courses in the educational use of ICT.

- Operations course.
- Negotiate contracts for operations.
- Ensure that the ICT contact and the ICT service have the necessary resources.

The advantage of settling in advance who does these tasks is that expectations on the individual are known, giving a good basis for planning and managing ICT services. Usually these ICT tasks are only done part-time by staff who also have teaching duties.

A business might well have two staff members working full time to operate 100 standard client machines for 100 users. In schools there may be a 30% position in total, divided among several persons, operating 100 client computers used by 320 students and teachers.

When schools have so few resources for operations, it is crucial to manage those resources well. Agreeing on who handles which tasks can make it easier to assess whether you need additional resources, or to reduce expectations of IT initiatives in schools due to budget constraints. An IT administrator with a good overview of the ICT tasks in the school is better able to ask for an increase in resources if necessary. There may be a need for increased resources to implement ICT-based exams, or a need for new equipment like digital whiteboards as teaching aids.

3.1.2 Expected time usage

We've created a table showing time spent on operations and maintenance. The table is based on the experiences of municipalities which implement a centrally operated Debian Edu of 9-10 schools with 250-500 client computers. Several things are not included in the table. Therefore extra time is required for projects where schools develop their own ICT solutions with networking and more equipment.

Role	***Operational responsibility***	***Time spend per school per week***	***Time spent in total for all schools***
Centralised operations staff	Monitoring, debugging and operation of 500 machines, for example, 10 schools with 3,200 students and teachers.	2-3 h(50 clients)	½ position(500 clients)
ICT contact at each school	Oversight of equipment, easy maintenance, and reporting of incidents and requests	3-4 h(50 clients)	1 position(10 schools / 500 clients)
Central ICT-coordinator	Assist in planning and implementation of educational and technical ICT work in schools.	1-2 h	½ position
ICT manager (principal)	Make joint purchases, and ensure compliance with the service level agreement. Schedule updates, or development of solutions	1 h	¼ position
Overall for a school	**50 client machines (concurrent users)**	**6 - 10 h**	
Overall for all schools	**10 schools, 500 client machines (concurrent users)**	**2 ¼ positions**	

Experience shows that the scope of work of the ICT contact is affected by the number of concurrent users. The term "concurrent users" is new to many. To illustrate with an example: A school may have 250 students but not more than 50 computers. Then a maximum of 50 students can use computers at the same time. This is much less than the total 250 users who have an account on the system. It is these 50 logged in users that provide work for IT service. The other 200 people not logged in give little extra work.

Therefore, it is common to calculate IT costs from the maximum number of concurrent users. Other calculation methods are also possible, for example when paying for proprietary software. But since Debian Edu has no license costs, the number of concurrent users is the most crucial figure for operating costs. To calculate costs from user accounts provides little or no meaning for a school.

For users of Debian Edu the cost difference to manage 100 or 250 user accounts is very small. There are a few exceptions. With 250 students instead of 100, some students may repeatedly forget their password. Therefore, it is wise to authorise the teacher responsible for the class to give these students new passwords.

If the school has 50 client machines, the ICT contact needs less time on their operational tasks than if the school has 150 clients. With more clients, the overall time spent on operations increases, but operating time per client machine goes down a bit.

Several municipalities have set aside 3-4 hours a week to the ICT contacts tasks at each school with 30-70 client machines. The Education Department in Oslo has set aside a day and a half per week, or a 30% position, to support 150 client machines. Experiences from other municipalities suggests that a 20% position is enough for the tasks of a local ICT contact when a school has 160 thin or diskless clients with Debian Edu.

In addition there are costs associated with centralized operations, ICT management, and building up the educational use of ICT tools in school subjects. One position is probably sufficient for the operation of 1000 client machines. As regards educational support, several principals have a 50-100% position in the school for this work. There may be a 10-20% position as an ICT contact and a 40-80% position as an educational support for the teachers. Many teachers perceive IT tools in schools as something new. Some principals wish to give more backing to the educational side by making teachers more confident in using IT tools across the different subjects.

3.1.3 Check list

We now give a check-list of what's needed to get a new service office up and running.

- Arrange people in different roles like IT manager, IT contact in each school, central operations and IT coordinator for all schools. It is important to distinguish technical operations and maintenance from teaching work.
- Establish the service desk such that every school has a service agreement regulating what is standard operating activities, and what is extra. It is imperative that ICT-responsible principals are a part of this process.
- Establish a system for handling incoming requests (a request tracker). All enquiries by email need a case number. Almost all enquiries from users or IT contacts from schools also need a case number.
- Ensure that ICT budget reflects the contribution necessary to ensure proper operation of school computer equipment and networks. The requirement today is that the ICT systems will be used for national and local tests with use of ICT tools with or without the Internet.
- Basically use the standard edition of Debian Edu with the same version on all schools. From this make the changes you want. These changes must be taken care of in a configuration database with documentation of the changes made. Version management can be used to save the changes and documentation.

3.2 Incident Management

The purpose of the ICT service is to prevent disturbances like shutdowns or software issues. Users will experience few problems with the ICT system if the ICT service has enough resources to handle operations, equipment and for enquiries to the Service Desk. Small or big problems will cause interruptions for users, so good handling of incidents is necessary.

In parachuting they call near-accidents "incidents". It is perhaps not quite the same in computer operations when something is not working. The purpose of dealing with incidents is to restore services as quickly as possible so that everything works normally. If something goes wrong, it must have the least possible impact on users. What is a "normal service" is agreed through an operating agreement describing the service level.

Statistics of incidents is important, especially if several people work within the organisation. When several people work together, it is easy to lose track of the work. Statistics will point out problem areas that must be addressed more thoroughly

than a quick fix from the service desk. For example, there may be many requests to replace forgotten passwords, so it may be wise to let the teacher change passwords for pupils in their class.

An operational disturbance is defined as:

- an event which is not part of normal operations and causes, or can cause, an interruption or reduction in the quality of the service.

Examples of operational disturbances may be:

- Programs
  - the office program (OpenOffice.org) does not start
  - the web browser (Firefox) crashes
  - the hard drive is full
- Hardware
  - the server is down
  - unable to print
  - unable to log in
- Requests
  - requests for information, advice or documentation
  - forgotten password

The examples show some of the most common operational issues. These are problems that prompt users to contact the school or the service desk. The ICT service must prioritize what must be handled straight away, and which problems need more time to resolve. To prioritize which problems need more comprehensive debugging, it is important to log all enquiries about malfunctions. Once one has an overview of the most common problems, appropriate actions can be taken.

3.2.1 Check list

We have made a short check list to ensure procedures and systems for good event handling are in place.

- The operator doing the debugging will report the status back to the ICT contact at the school and/or the user.
- The system for logging events must be available and working (both technically and functionally) for those working with event handling in schools and at the service desk.
- The event logging system must be used for virtually all operational events.
- Statistics of the log of events should be made periodically. The statistics can be used to identify and eliminate recurring problems, which are irritating to users.

3.2.2 Planning and implementation

To set up a workable system for logging events requires something more than installing the system. Everyone in the operations department must use the system. Those reporting errors must also receive feedback by email with a ticket number. This requires significant efforts in configuring the system for event logging. In addition, one must ensure basic user training for those who receive the requests.

Large and comprehensive plans are not required to implement proper event handling. Event handling is a completely standard task for those who work at the service desk or as ICT contacts at the schools. Setting up a computer tool for logging events may require up to a few weeks for a correct configuration, and users may also report events via e-mail and by phone.

The user interface to the logging system is relatively self-explanatory, so it should not take too long to get started. Daily use of the system will get users comfortable with what should be logged. It is crucial that everyone in the operations department uses the logging system for operational messages.

3.2.3 Activities related to operational events

To get an idea of activities done following a reported event, we use an example.

A user contacts the service office with a problem, and reports that printing is not working. Operations logs the event immediately after the call is completed. A case is opened for the issue, and automatically given a case number.

Operations at the service desk make a quick analysis. Has the spooler stopped again, or is it something else? Is the paper or toner missing? The operator examines the spooler and sees that queue has filled up. She deletes the queue and tests whether the next job is printed.

This time the print queue fills back up again. Operations contact the school's ICT contact asking to check whether the paper tray is empty. This is listed in the event log. The ICT contact replies that they have refilled the paper tray, and printing is normal. The case is closed, and is noted in the system event log.

If printing had not started again, the toner might have be missing or there might have been a printer error. If there was an error, operations would have to escalate the issue. This means that someone other than the operator or the ICT contact is needed to resolve the problem - in this example, a technician who can fix printers.

This example shows the whole workflow that needs to be investigated to get a printer working again. If a printer does not work even after checking that paper and toner are available, the issue needs to be escalated. The operations department must call in an expert to fix the problem - this time it was a service technician for printers.

What was wrong and what the fix was are noted in the event logging system.

3.2.4 Roles

A variety of roles are involved when the ICT service deals with reported issues. In the example above, the school's ICT contact and the operator cooperate to solve the printing problem. Had the issue been more difficult, they would have had to call a technician. If the printer could not be fixed, a new one would have to be purchased. If the school needed to buy a new printer, the ICT managers might need to arrange payment. In many organisations, the principal has the last word.

In short, it is easy for many people to get involved when something does not work. If possible, problems should be solved on the spot, trying to avoid including unnecessary people. Escalating problems which could be solved locally quickly becomes costly. Many enquiries are easy to deal with there and then, but other requests involve more complex problems which involve more people. If additional or external help is needed to solve the problem, this must as a rule be clarified with the operations manager. The important thing is to be aware of these points when handling operating events, so as to use resources appropriately.

3.2.5 Key points

We have sat up some key points for handling incidents. These points can be helpful in evaluating whether or not things are going well by using measurable and well-defined requirements. Such measurement points are:

- Total number of operational incidents.
- Average time from receiving an inquiry to when the issue is resolved, classified with codes (a well organized operation department has codes for different types of events and errors).
- Percentage of incidents handled within agreed response time (as agreed in the service level agreement).
- Average cost for each event
- Percentage of incidents solved by the service desk without escalation
- Events per client machine (workplace)
- Number and percentage of incidents solved by the operations center without the need for visits to school

3.2.6 Tools

A number of tools can make it easier to handle operational incidents.

- Automatic logging
- Automatic routing of events to the right persons
- Automatic retrieving of data from the database for configuration management
- Phone and email are used in conjunction with tools for registering requests and incidents.

3.3 Problem Management

Problem management is an "investigative" process. Known bugs are most often handled directly by the service desk. This is the most common form of event handling. To investigate unknown errors requires both common sense and instinct. Good operating people use instinct to go straight to the problem, find the solution and restore service as quickly as possible so that everything works normally.

Problem management is;

- Problem management
- Checking errors
- Proactive control to prevent problems
- Identify error patterns, using information from, for example, event management

Problem control

- Identify problems
- Classify problems
- Examine/research problems

Error control

- Identify and register known errors
- Find temporary solutions if possible
- Contacting those with responsibility for Change Management to remove the error permanently

Proactive control

- Identify and solve problems and errors before the incident is reported by users.
- Use logs and information from event handling to see how problems may arise

3.3.1 Procedures for problem management

The Skolelinux / Debian Edu manual is a comprehensive collection of solutions for solving problems and configuring systems. Everything is on the Debian wiki pages. Solutions are maintained with the help of staff in schools, municipal ICT services, professional individuals and volunteers. See links to the English pages: https://wiki.debian.org/!DebianEdu/Documentation/Manuals . The pages are being translated to Norwegian bokmål. We are working to link the pages to bokmål too.

The Wiki technology has proven to be a great success for maintaining catalogued information on the internet. It's easy to contribute to and all changes are logged. It is also possible to import OpenOffice.org documents, and export documents as PDF.

3.4 Configuration Management

The resources spent on IT systems in schools must be handled in a financially prudent manner in order to control the services used and the equipment / infrastructure. The equipment, software and services have a whole range of settings - this is configuration, or a logical model of how infrastructure and services are set up.

To manage configuration it must be identified, saved and maintained. One must also be able to keep track of different versions of the configurations. We call each part of a setup for a Configuration Item (CI). A configuration file may, for example, ensure that certain users have access to a few printers in the network. Another can make sure you get a buffer on diskless clients.

An updated database for configuration management is essential to ensure rapid and controlled handling of operational issues, or changes in the layout of machines, programs or services.

3.4.1 Planning

It takes planning to set up a database for configuration management. One must decide in which areas to use the system, the objective, policies and processes for storage and maintenance of configurations.

- Identify and select a structure for configuration according to the important parts of the ICT infrastructure. Configuration owners, name tags (attributes), dependencies, and relations between configurations all need to be considered.
- Only approved configurations are managed in the database through the lifetime of the system. Control over access to the configurations can be done with group permissions, and can be done through the process of Change Management.
- Status logging - keeps track of the condition and status of the various subsystems. This applies throughout the lifetime of the service, software or hardware. There may be a configuration in production, disconnected or discontinued.
- Checking and revision. Each configuration must be checked to confirm that the correct information is stored in the configuration database (CMDB). This is followed up with periodic reviews to ensure that the database is up to date.

As we see, there is a lot of planning needed in order to have configuration management in the IT system. The purpose of planning as part of IT operations is to ensure that systems are fixed quickly when they go down. With a good configuration management, it is easy to replace a defective machine with a new one. The configurations can be quickly transferred to the new computer and the IT system functions just as well as before.

3.4.2 Management of Configuration Items (CI)

A configuration item is a part of the infrastructure. It is normally the configuration of a service or a program. Some times users want to change how a service work. One need to keep track of the configurations if changes are made.

To get this down to earth we can imagine the configuration of the printer server. You want to add a new printer to the computer network and will add this to the printing system CUPS. When changing one configuration through a web application or via configuration in KDE. CUPS config file will change, and you must restart the printer server again. This can be done in KDE tools or through a web application. The modified setup file is copied to a directory where the file can be handled by a version system.

Of many different choices there are a few common ones. This is if a service should: run, stop, terminate, start, be interrupted or taken out.

One should be cautious in changing configurations without a proper plan. It is easy to forget what you have done on a server or a PC. Therefore it is important to document the changes made in a change log.

3.4.3 Planning and installation

The configuration of the computer network is connected to the architecture. Much of the planning is done with Debian Edu. This is because it may take both 3 and 4 weeks to set up servers with corresponding service level with Windows server, RedHat or other GNU/Linux distributions. Debian Edu takes this with 1-2 hours. If you want a fixed IP address for the network a professional uses ½ hour extra on this. This is because web services are set up with reusable names.

What then must be planned is which additional user program to use, and which subsystems should interact with Debian Edu. It may, for example. be that the school has an electronic whiteboard.

3.4.4 Check list

We have made a list of activities and solutions that are important in good configuration management.

- Establish a version-controlled area for saving configurations for all servers and selected workstations and laptops. Git and SVN are often used for this. Remember to take daily backup of the area, and make sure to save all changes in configurations.
- Use an electronic system for taking care of recipes explaining configurations of different type machines, the network and services. Such recipes contributes to others who help or take over operations can read up on what is done. A wiki can be suitable for this.
- Use one specific version of the operating system and software on all machines. This is to avoid maintaining many different versions of the software. Ensure that the software is well tested. Therefore, it may be wise to wait 6-12 months before adopting latest edition of a program.

3.4.5 Relations to other processes

Management of configurations are closely connected with the handling of problems and if the systems are available. If printing stops to often, it may be that a configuration change solves the problem. It may, for example, be to establish a routine for deleting the print queue and restart the print service anew.

The aim of the changes you make in the configurations are usually to increase the availability of services or programs. It may also be to restrict access to certain programs or services to specific times. To achieve this, one must reconfigure the service. In addition, it may cost money beyond what was agreed on as service level or capacity of the system.

The examples show that the managing configurations engages a number of other areas. Therefore there is much to gain by putting in place good practices for managing changes in configurations. Also automation is advisable if you want greater stability, or access to certain services in specific periods.

3.4.6 Tools for configuration management

As mentioned under Check list one may use

- Saving the configuration files in a version-control system, for example subversion.
- Wiki for storing documentation of setup and wizards
- Use a common directory for operational documentation on the internet, maintained by Skolelinux/Debian edu staff in the schools.

3.5 Change Management

Many ICT services are not clever in handling changes in ICT systems. Leading to many disgruntled users. Surveys in the public sector in Denmark show that operating costs go down when you have good control on the changes. Therefore, it pays to involve users with training and participation related to the changes made.

Change-messages is entirely dependent on proper processes. This applies regardless of whether the changes are small or big. Therefore it is important to have in place the right people when making changes, both to give training and to have people to answer questions. This becomes especially important when adopting new releases of software and services. This is independent of whether one uses free or proprietary software.

Change Management should ensure that all changes are made in a standardized and right manner. It is important to anchor the decision about amending at the appropriate level in the organisation, Standard changes can often be pre-approved when they are done a few times. But major changes will often involve a higher decision level between school management and operator.

The reason why the management should be included is that an upgrade will often require training of users. It may be upgrading to a new browser or a new version of office software. This can quickly lead to a half day training in what is new in a program. Such changes must be agreed with the management. The changes must also be done without the other parts of the system stops working.

Those with responsibility for approving changes receives a so-called change message or RFC (Request For Change). When you have a RFC you can assess whether the change should be performed. Many times you have to clarify with management if optional changes should be made, and if so, when it will happen.

By changes one must also cooperate with the school's ICT responsible. One must ensure that changes occur when it fits with the schools plans. To implement significant changes without Change Management can lead to much dissatisfaction and additional inquiries to the Service Desk. This would provide significant extra work without this being planned. In addition, it may lead to a change that would soon be rolled back. You fast get twice as much work without ending anywhere else than back to start. Had one made the necessary approvals, may the change be done in a planned and straightforward manner.

Change Management is done to avoid more extra work than what's necessary. Making changes obviously requires more work, but you will get less extra work on the changes planned. One also avoids the need to roll back changes, because problems arise where users are unprepared for substantial changes.

When you for example update the entire system to a new version, make sure that everyone is informed. One must look into whether those affected by the change need training. The right professionals must prepare it all, so there are no surprises.

All responsibility must not land on the person responsible for managing versions of software, the release manager. Release handling is a process which preferably should work with changes that contains many minor changes. This usually happens when rolling out new systems and services, or the upgrading of the entire system to a new version.

3.5.1 Activities

- See change message, or RFC (Request For Change) above, and check it also has got a unique number.
- Prioritize and categorize the changes
- Remove not possible changes. This can be done by marking them as not possible.
- Give feedback to the one giving the change message
- Make sure you have a Change Advisory Board, where the change is dealt with, discussed and evaluated. This consulting group can be selected ICT contacts and operations personnel with long experience.
- Coordinate changes with the Release Management which handle different versions of applications and services.
- Look over and finish the changing message (RFC)
- Remember to save modified configurations in the repository for configuration files.
- Reports

Even what may look like a small insignificant change message can have major consequences for if the change is implemented. We have examples of schools that have a stable Debian Edu network where all the programs work. A test version of a popular program crashing constantly, is installed, and Debian Edu get blamed.

An example is schools that have installed the test version of the latest OpenOffice.org before the program was finally finished. Several thought it could be fun and try out. The problem is that the test editions are usually released to find errors and instability in applications. They are not intended for production use

In production, the general rule is that you don't install test versions of software. Most operators recommend using the next to latest version of a program intended for production. After 6-12 months are usually the worst errors picked out of a new main version of an application.

It means one often wait until summer before updating to a program that were reissued just before New Year. This fits well with the school year. The alternative may be instability and irritated users. Therefore the advisory group plays a key role when done small or large changes.

3.6 Release Management

Release handling is management and planning activities preparing for wanted changes. The changes can be small or large, where large changes can consist of many smaller changes. Release management goes on before initiating the actual job of installing software and hardware into production.

First the planning and testing of new releases are carried out. Then it all is rolled out it into production. Deployment is part of the infrastructure management. The procedure is to implement what is planned, tested and is ready within the systems for Configuration Management. Once everything is planned, tested and configurations are stored, then roll out the solution in production.

Usually, many service providers and suppliers are involved. This applies both to the acquisition of machines, the software used, and the recommended configurations. Good resource planning is crucial to package and distribute a new release in a good way for users. Cutting corners in this area can lead to equipment that doesn't work, or that goes unused because of deficiencies in the installation.

Release Management takes a comprehensive approach by the change in a service, and ensure that all parts of a publication is seen in context. This applies to both technical and non-technical factors.

3.6.1 Basic

As you can see, for computers, software and network to work as planned, release-management is crucial. Proper handling of releases prevents disruptions. New releases or changes can be introduced while operations continue as normal, without interruption or reduction in quality.

Implementing changes or new releases can be compared to building a new road. Cars must still get past even if you build a new road on top of the old. Good signs must be in place. One must also have the necessary resources to rebuild the road. If you lack the resources to make changes, it's better to let it be.

Some might think that proper release management is boring as one doesn't get to implement the latest version every time something new is released. But often the operations department lacks the resources to handle a flood of complaints should an upgrade fail. High uptime requires established technology, as said by Linux expert David Elboth in the Linux Magazine (1/2004). He writes:

- The more you demand of the system the more stringent are the requirements of the individual components. High requirements for uptime results also show that the choices you are left with are old technology. Only empirical data over time can say anything about downtime. We have all noticed how far behind are Red Hat and !SuSE with their server products.

To get few complaints, with a stable and reliable environment, requires solid release management. Alternatively, a bunch of complaints and dissatisfied users emerge, caused by installing insufficiently tested cutting-edge software. Amateurs have a tendency to underestimate the consequences of software upgrades. If something works fine on your home computer, it does not mean that this will work in a wide network with 500 client computers and 3200 users.

3.6.2 Definitive Software Library (DSL)

A software archive in an operational context is a collection of original copies of the software in use. If you use Skolelinux 2.0, this is the software package. The phrase software archive is used differently in some other contexts, especially among programmers. When it comes to operations, we would be talking about the original software package of a particular version which is used for the installation.

By using free software, the software archive may be Skolelinux 2.0 plus the extra programs you have added from various sources. There may be certain versions of Macromedia Flash, Java and decoders which make it possible to run national tests in the browser, or to watch broadcasts from a national TV station.

If you plan to upgrade to the next version of Debian Edu when released, this new version shall be the main program archive. The new archive shall also include appropriate versions of all additional applications beyond Debian Edu.

Set-up files customized or created locally by the operations department are not included in the main program archive. Configurations are saved separately in a version-control system or database.

3.6.3 Database for configurations and hardware

As mentioned in the chapter on configuration management, you must create a database or a version-controlled directory to take care of set-up files. One should also keep track of all computers, what kinds of machines are in use, performance, and unique standard addresses on the network cards (MAC addresses).

There are many reasons to have an overview of the equipment. One of the main reasons is to keep track of how many machines are in operation, how many are not in use and how many are being repaired. Another reason is planning for upgrades.

3.6.4 Build management

A variety of applications in addition to browser and office suite are installed in schools. Educational programs for learning, browser plug-ins, and programs for multimedia are needed. The systems also have network set-up and changed settings in specific programs. When you have many servers and perhaps thousands of clients, the need for effective tools for deployment, soon makes itself felt. Such tools are standard in Debian Edu.

Build management is about ensuring that you always install the required software packages, services and proper settings both of individual programs and for the network. Many people have heard about the so-called "images". One installs the operating system with all needed programs and configures the network. Then one uses an image program to make a copy of the hard disk. This "disk image" can then be copied to other computers.

It is not necessary to build such disk images. Debian Edu is based on Debian which has an excellent package management system. There is no need to compile applications, as ready-made packages can be installed directly from the Internet. It is enough to work out what changes you want to the default set-up of Debian Edu or the main program archive in use. Then you make one or more scripts to run on each machine that get everything installed and set up.

For most situations, scripting is an easy way to "build" and roll out programs and configurations. But there are situations where building disk images may be the solution, e.g. for installation on many laptops.

As demonstrated, handling the build process is about facilitating deployment on many computers. In exceptional cases, this may involve building a tailor-made Debian package. But in most situations, everything is ready-packaged. Then you must write and deploy a script which installs additional programs and certain settings. One can also create disk images if you have many similar machines, such as laptops for all students.

3.6.5 Testing

It is essential to test new applications, configurations, and new services before they are put into production. Several schools have experienced instability when they have installed software without making the necessary adjustments. Therefore it is crucial to test changes in configurations or new versions of the software before the change is made on all machines.

Testing generally takes place in three steps.

- First, do an installation of the changes on a test network. This is technical testing to check that everything functions in a system without users. Take care to include all changes in the configuration files.
- When you are sure that everything works on the technical side, try installing the solution in one school. It is very important to book the testing with the school's ICT contact. Users must also be fully briefed on changes made for the sake of testing. Take care to preserve current adjustments in the set-up files, which may have been made in the course of normal maintenance.
- When you are sure everything works, you can roll out the solution to all schools. It is easiest to create a script that simplifies upgrading of software packages, services and configurations.

3.6.6 Fall-back solution

Much can go wrong during a new installation or upgrade. Therefore, one must have ready a fall-back solution. This lets one quickly get back to the system as it was before the upgrade. In technical terms, this is called roll-back.

When rolling back it is absolutely essential to have the previous version of the software archive and configuration files ready. This means that you can install for example Edu 1.0 in under an hour, and put in place the appropriate configuration files.

But roll-back takes time. Therefore, it may be prudent to have a server ready with the previous version of the software, the right configurations, and a recent copy of the users' home directories. This server can quickly replace any machines on which the upgrade does not work according to plan. Having server machines in reserve can ensure high availability even if something goes wrong.

3.6.7 Advantages and possible problems

The advantage of having records of the software in production can't be underestimated. Many rely on having the software on their respective CDs and DVDs. This a inefficient method of distribution. To save time and trouble all the software in Debian Edu is available online.

Your operations department can create a copy of the Debian Edu archive on a central server. From here, all the software can quickly and smoothly be installed on other machines. The advantage is that your ICT service has a constant overview of the versions of the software they have made available to schools. This also prevents the installation of software that has not been reviewed by the Change Management.

There may be considerable problems if you do not maintain the software archive and configurations. It might happen as well that one makes mistakes with a configuration or software package. Then this gets rolled out to all machines. In addition, some schools may install insufficiently tested software or use beta releases in production. So one must have good processes and have someone to take responsibility for maintenance of the program archive and configurations.

It may seem like one needs a lot of extra things in place in order to install and maintain the services and programs that are in use. However, if you skip the tools that provide management of upgrades, you give yourself a lot of extra work. The ICT service must spend a lot of time on manual installation on each machine. The danger of making mistakes increases. When things do not work you get disgruntled users, and much time is spent fixing problems.

Many of those administrating big IT systems have inadequate plans for the upcoming changes. Some have no plans at all, but just upgrading the software to the latest releases. Changes made can be perceived as problematic for some users, because functions they are comfortable with could change location on the user interface. For operations it can go completely wrong. For example when they tried upgrading from older to newer version of Windows in the Arendal municipality, mostly everything stopped working. The IT department said they had several computer programs that were held together with "wire and tape." It took half a year to clean it up.

3.6.8 Planning and implementation

The reason for planning before implementing changes is to avoid weeks or months of delay due to problems. The time used for planning is quickly regained because one avoids additional problems. There will always be people who say they have had no problems with ad hoc changes in the systems; but closer examination reveals that there are problems after such changes, they merely don't get communicated.

In our eyes, ad-hoc solutions are only a detour through changes, and only an emergency measure. An ad-hoc solution is like a temporary repair with "wire and tape." One must in due course clean up such solutions to ensure stable operations without constant surprises. Skipping a planning phase leads to many more ad hoc solutions, and several operational problems when changes or upgrades are done. Therefore it is essential that professionals and management understand the value of a good planned process for changes.

Therefore, we recommend that you convene a meeting for planning, and make a stepwise plan for changes in the system. A stepwise plan will naturally vary according to the change. Upgrading the OpenOffice.org suite is quite different from upgrading the whole system. When upgrading to a new office application, a 2-3 hour tour of the office suite may be enough for the teacher in each school. When upgrading the entire system one must both provide user training and test that the technical details work as intended.

The main point is that it is straightforward when it comes to planning and implementation. Studies show that those who plan properly and ensure that people have the right skills, have lower operating costs for the operations.

3.6.9 Activities

It is crucial to plan new releases. Most modifications of the system should be clarified with the management. The following list of activities is designed to support the upgrades in a planning and implementation phase.

Tasks	Details
Prioritization of the release:	Check if the necessary decisions are made before a change or upgrade would be deployed.
Definitive Software Library	Ensure that the appropriate software packages to be installed are in place in the definitive software library.
Configuration database	Be sure to have in place all configuration files. This applies both to those who are in use, and the new ones supplied in systems to be changed or updated.
Build management	All scripts and systems used to deploy or create disk images must be in place.
Testing	First, run trials on the test equipment. When this works without any problems, it can be tested at a school. The school must be fully informed about this, and fully aware about trying out new software. When one is sure that everything works, you can upgrade for all the others.
Fall-back solution	Even with extensive testing, new releases may go wrong. Therefore it is essential to have a fallback. The easiest solution is to spare the old installation with its data on a separate server machine. Such a machine can be plugged in again if the change or upgrade does not work.

3.6.10 Tools

As seen from the activity list, one needs several tools to keep track of different releases of software, services and hardware in the system. Some of these tools were mentioned previously. But we repeat this anyway:

- Debian tools for the definitive software library
- Configuration and hardware asset database (subversion setup files and spreadsheets providing an overview over all hardware and their respective physical locations)
- Build management (the system which builds Debian packages)
- Hardware for testing and backup-solution

3.6.11 Relations to other processes

Release management fits directly into the core of the ICT services. It is about implementing appropriate security updates, changes in the services or upgrades of computer software. Requests for new releases may be due to operational problems or wishes for new software. An assessment on whether the change is necessary is done prior to committing the new release.

If the change is straightforward one would make the necessary changes in the configurations and make the application packages ready for deployment. This would be tested, and one would have in place backup solutions. When the changes are made, one should perhaps alter parts of the operational routines. It would be easy to see how change management affects all parts of the operational support.

3.7 Tools for operational support

The first thing you should ask yourself: "Do we really need software tools?" If that's true, it is crucial to examine the options thoroughly.

Taking a glossy brochure, and listening to sales talks, one is totally dependent on such tools. But good people, good process descriptions, good procedures and job descriptions are a basis for good service management. The need for, and how complicated the tools are, depend on the organsation's need for computer systems, and the size of the organisation.

In a small organisation, will a single freely accessible database be enough for logging and management of events (request tracker). But in larger organisations will one almost certainly need a sophisticated distributed and integrated tools for service management. It means linking all processes to a system for event handling.

Although tools can be important, they are not important in themselves. It's the tasks and processes to be done, and the information needed which are important. They will provide the necessary information to specify which tools are best suited to support the operations. Here are some reasons why one may use software for operational and service management:

- increased demands from users
- lack of ICT knowledge
- budget limitations
- organisations is entirely dependent on the quality of service
- integration of systems from multiple vendors
- increased complexity of ICT infrastructure
- emergence of international standards
- Extended scope and changes in ICT

Automatic tools allow:

- Centralisation of key functions
- Automation of functions in the service delivery
- Analysis of data
- Identification of trends
- Preventive measures may be implemented

3.7.1 Type of tool

In this chapter we have proposed a number of tools to improve operational support. Here follows a summary of the tools:

- Debian tools for the definitive software library
- Configuration and hardware asset database (subversion setup files and spreadsheets providing an overview over all hardware and their respective physical locations)
- Build management (the system which builds Debian packages)
- Hardware for testing and backup-solution
- Request Tracker
- System for monitoring (Munin)

As the operations department gains further experience with systematic routines, additional and different types of tools will be developed or acquired.

3.7.2 Evaluation criteria when selecting tools

Although it's been used large amounts of money on creating evaluation criteria for software, the result is only experience-based guidelines. There is no final answer to what's good or less good software. As much else, it revolves partly about taste. Different solutions do the same job just as well, but may have quite different forms. However, here may some rules of thumb be useful.

The main evaluation criterion is whether one needs to execute a job at all. Many IT tools are absolutely perfect and work without error, but might solve tasks that are not needed to be fixed. So the main criterion is whether it resolves the right problem, and if at all it is necessary to do anything.

- So the first thing one asks is whether the tool is needed.

If it turns out that one should complete a task, the solution might be as simple as to run some commands manually. The simplest way is best. But when one gets many machines to operate, automation becomes crucial. It's too much work to log into 20 similar server machines to do a security upgrade. Then automation is the thing.

- So here one must ask whether the tool is useful to solve the task
- Then one must ask whether the tool is usable.

There are often a wide range of programs and procedures to solve a specific task. But some problems are solved completely differently when maintaining 500 computers and 11 servers, than when fixing your home PC. An example might be tools that allow the teacher to see the desktop of each student on his or her client machine. The teacher can stop and start programs for all pupils, and prevent individual pupils from using for example IMs when this interferes with school work.

Regarding the choice of operating tools, it's about automation and simplification of operational tasks. It is about making and reducing manual work to a minimum. So the motivation is to just to maintain the automatics. Also here it is possible to make things easy, which can be a considerable job to fulfill.

As you can see, it is not easy to set up good criteria for the selection of operating tool for large installations. Mainly this is because software developers often lack experience in the operations of IT systems. They are known only to create new things, but to create good and relevant tools for operations, requires many years of experience.

Some general operational tools have not been replaced the last 20 years. But the products used may have been replaced. Also some software may, in a few years time, be irrelevant to use. Therefore, one must be ready to get some training in new editions of the applications used for operations, and in upgrades and changes in end user software.

3.7.3 Product training

Thorough user training makes that a lot of support issues can be resolved informally in direct communication between the end users themselves. Often, training costs as little as 1% of the total operating costs. It is well worth spending a little more on training. The effect is very positive. The same applies to proper training of the ICT contacts at schools, and the operators. Training of ICT contacts to use simple systems for password change, error messages, etc.. will provide better quality of calls to the IT service.

Education and product training are in Norway regulated according to the Labour Act (§ 4-2)

- Employees and their union representatives will be kept informed of systems used in the planning and implementation phases. They should be given the necessary training to familiarize themselves with these systems, and they shall take part in designing them.

So in short terms it can be advantageous to increase efforts in training, which will improve ICT service quality and provide a significant cost reduction. This is because users and IT contacts become more confident and better in helping each other. It should also be noted that the transition to new software can also provide an opportunity to simplify some of the operating practices. Simplification can reduce the requirement for product training.

3.8 Planning at the start of the implementation of service support

A growing number of organisations sees the necessity of service control. It is often a usual practice to base decisions on historical and political considerations, rather than the current organisation's needs. Therefore it is important to ensure that the management commits to participating in and understanding the working methods in the organisation, and go through the existing processes and compare those with the organization's needs and "best practices".

3.8.1 Implementing service support

Health check

3.8.2 Feasibility study

3.8.3 Determine current situation

Health check

3.8.4 General guidelines for project planning

Business case for the project

Critical success factors and possible problems

Project costs

Organisation

Product

Planning

Communication plan

3.8.5 Project review and reporting

Progress

Evaluation of the project

Supplementary work.

Reviewing to check the compliance with the quality parameters

Reviewing in regards to key factors

4 Service Delivery

The main purpose of service delivery is to ensure proactive operations and that the ICT services deliver appropriate support for users. The purpose of service delivery is to focus on your organisation's needs. It is active learning, with use of ICT tools in the different subjects, the school needs. This chapter describes in order:

- Service level management
- Economy management
- Capacity management
- Capacity planning
- Access control
- Operational continuity

4.1 Service level management

Service Level Management is often shortened to the acronym SLA. Managing the the service level is about the quality of the operational services, measured in relation to what is agreed in a contract. There are definitely concrete figures for availability, response times, support, error correction etc.

The objective is to have control over the service level and improve the quality of the operational services. By repeating rounds the quality level is determined, monitored and reported. The purpose is to improve the contact between ICT administrators and users, to get an ICT service, to the agreed quality, delivered.

It is important to understand the different types of SLAs. One can choose from many types of agreements. The three most common types are:

- Agreement per service for all customers
- Agreement per customer for all services
- Agreement per service per customer

All SLAs are to be administerated, reported and maintained. It can quickly become confusing and produce much work that does not provide a particular benefit. The purpose is to get an agreement that helps to improve the quality of service. Therefore it is useful to think carefully about that, when the agreement is made. Here is an overview of what is important to make sure about when you create an agreement for the service level management.

4.1.1 General checklist

- The agreement between the user and the operations of what's actually being measured. This must be seen from the users' perspective and not the ICT services perspective.
- Measurement and clarity about the metrics included in the SLA
- Decide realistic targets for the service level (there is no point in promising more than one can keep)
- Continuous focus on the control of the service - monitoring and periodic reporting of results achieved

4.1.2 Planning

It is essential that the operations center has the technical capability to measure the values included in the SLA. This must be taken into account from the beginning.

Furthermore, it is important to define the services where one is dependent on subcontractors and therefore can't provide service guarantees, or relies on a similar agreement with the subcontractor. The definition of dependencies is made because it should be clear who should rectify the problems, and to avoid never ending negotiations until the error could be corrected.

Level of service may be different for different user groups, or during different periods of the school year. For example, there may be differences between teachers and students, or a higher service quality when carrying out exams. Dialogue with all relevant users is important to ensure measuring of what's most important for each user group.

4.1.3 Implementation

A service catalogue with all services included in the SLA must be prepared. A service will often be an application (program) in this directory. It will often be different requirements for different services, and it will be reflected in different objectives in the agreement.

Establishing and continually adjusting the users' expectations can't be overestimated. Often users have exaggerated expectations to the system and the services included. ICT services' responsibility is to adjust expectations down to realistic levels before the service-level agreement (SLA) is signed. Operations management must also ensure that all users actually are notified and know about the expected service level through the agreement.

For the structure of the SLA, see section in the service level agreement.

4.1.4 The operational situation

Monitoring of the actually achieved service levels, and reporting back to the customer, are essential to preserve a good relationship between the Service Desk and the users. Format and levels of detail for reporting, should be dealt with in the SLA.

It must be held periodic, for example quarterly or semiannually, meetings with the client. These meetings should result in concrete plans for the next period and, possibly, agreements for the implementation of new services.

4.1.5 Content of the Service Level Agreement (SLA)

4.1.5.1 Introduction

Name and contact information for the Contracting Parties, description of the services included, duration of the agreement, responsibilities between the customer and the supplier.

4.1.5.2 Service time

During which time period would the agreement be valid (like from Monday to Friday, from 8:00 a.m. to 4:00 p.m.), any special requirements at defined dates and times (for example exams) and routines to order an expansion of the service time limits.

4.1.5.3 Availability

Access to the services. Is best measured as the period of time when one or more services have been unavailable, for example a calendar month. Different levels for different services may be agreed, for example depending on the degree of importance for users.

Important to emphasise that this is availability within the agreed period of service, not the overall availability all day, all week and all year round (called 24/7/365). For example, it may be agreed that the system should be available between 8:00 and 18:00 on workdays, after that and on weekends it is more uncertain whether one can use the computer system, unless otherwise agreed.

Availability also means getting support via phone or email. For example, whether the Service Desk can be reached between 08 and 16 during the day time, or if it can be reached the whole day, or in the afternoons and evenings, or even during specific weekends.

4.1.5.4 Stability

Is often measured according to the amount of downtime in a period of time, or the average time between downtimes. One can also measure the time it takes the system to come up again after downtime.

4.1.5.5 Support

Often measured as response times by phone (for example 1 minute) or email (for example 30 minutes) to requests from users. When the operator gets a request for support, the message will be categorized by severity with a time guarantee for answers. There may also be an agreement about how quickly error correction will start, which will depend on what kind of inquiry was received.

The support is also about when during the day or night one can reach people. Should support be available during school hours between 08 and 16 o'clock, or should one also have support throughout the evening or on weekends. Some will have support also on certain holidays.

The period when support is available is usually in the SLA. It is also agreed what support will be available, with a fixed price, and what must be resolved additionally on an assignment basis. The agreement regulates the process of handling enquiries, both what to fix, and when this will happen.

4.1.5.6 Capacity

Can be measured as the average response time by certain operations in specific applications. Will measure user experience of the system.

4.1.5.7 Change management

Measurement for the management, approval and implementation times of change requests from the users.

4.1.5.8 Security

Can be measured as the number of ascertained security incidents in a period. It is very important to be clear on each user's responsibility to ensure that warranties will apply.

4.1.5.9 Billing

Prices, times for billing and settlement provisions.

4.1.5.10 Reporting and follow-up

Description of rules and periods for reporting of measured service levels. Regular meetings are recommended, for example quarterly, to go through the report and plan ahead.

4.1.5.11 Sanctions and possible incentives

Rules for price reduction if the agreed service is not met. Escalation procedures and rules for cancellation of agreement by continuous violations of guaranteed service level. Possible incentives for achievement or better than expected service.

See Appendix A for SLA.

4.2 Financial Management

Organisations rarely have a full overview of their ICT spending. A 2001-survey of Norwegian municipalities showed that only 1 of 8 municipalities had an ICT budget. It is probably not better for schools. Putting in place an ICT budget is important. Often users think they pay too much for a service they are not happy with. This often creates conflicts between users and the ICT department.

It is very useful for both the operations center and the users to document the real ICT costs. Without this, it is difficult to budget appropriately. And mostly, it is difficult to make a cost/benefit assessment of existing ICT solutions. The rector should know the ICT budget as well as she would know the salary budget, or the budget for the teaching aids.

There are three major key processes related to financial management of ICT services:

1. Budgeting
2. Accounting
3. Billing

4.2.1 Budgeting

The objective of the budget is to make a realistic estimate of the expected ICT costs. Budgeting usually contains various alternative solutions. It applies both to equipment and software, and the level you aspire to. The budget is the starting point for subsequent budget negotiations with the director of education and/or politicians.

Budget must include both personnel and equipment costs. Some organisations only count the cost to buy equipment, omitting as much as 60 - 70 % in personnel costs for the operation of an ICT-solution. One must also get all of the equipment.

There are examples of municipalities forgetting to count the cost of power connectors and computer networks in schools. Then you have forgotten about 2000 NOK (10 NOK = 0.85 GBP/1.18 EUR) per client machine. For 70 new computers, we need about 140,000 NOK for computer networks and power.

Alternative solutions are also important to include in the budget. This applies both for the operation and the equipment. Today there are several vendors who specialize in the operation of computer equipment in schools with varying prices and quality. The number of simultaneous users, and type of machines and software to be maintained, is important.

If one would like to have laptops for all teachers and students one will easily get 5-6 times higher costs than if one had desktops with three students for each client machine.

4.2.2 Accounting

Accounting will mainly consist of invoices for purchased equipment, cabling, repair, operations and extra services. When the accounting period is over, it is important to go through the numbers and compare this with the budget.

4.2.3 Planning the accounting and billing

Not all municipalities have accounting systems that show ICT costs detailed by school. There may be practical reasons for that, such as discounts and similar that the municipality gets centrally. Therefore it is important to do some planning so that you get an overview of what were the costs for operations and procurement when the accounting is assessed against the budget.

Some organisations may have cumbersome and costly accounting procedures. You quickly get extra charges if you pay bills late, or there are many who must approve a payment, for instance. So it is important to agree on good billing practices in procurement and operations in order to have control, as well as to handle payments on time without long decision processes.

4.2.4 Implementation

The payment method is regulated by the SLA. When it gets to the accounting system, one must agree with the finance department for a convenient way to get out the reports, in order to get the necessary accounting overview of ICT costs without it taking a long time to be generated.

4.2.5 Daily operation

Regarding contracts one will usually have a fixed monthly billing consisting of a fixed amount and possible additional services. Billing is done from the accounting office based on the current operations' contracts, and the extra services performed. It is important to have good and frequent contact with the accounting service based on the tasks carried out for the customer.

4.3 Capacity Management

Capacity planning is used to ensure that all parts of the ICT solution have sufficient capacity to safeguard users' requirements. This includes:

- Monitoring the performance of ICT services and their related infrastructure

- Configuration of the systems to ensure they are optimally utilised to what the users actually do
- Understanding the user needs and planning for possible changes in the systems to take care of future needs
- Resource planning in cooperation with the budget officer
- Preparation of a capacity plan to ensure delivery of operations in accordance with the agreed upon service level

Capacity planning is all about balance:

- Costs against capacity. The budget limits what kind of possible solutions one can implement
- Supply and demand. The systems must have the capacity to handle the demands set by the users

The objective of capacity planning is to avoid surprises.

4.3.1 Monitoring

It is essential for good capacity planning that the systems are continuously monitored to obtain the necessary data. Typical data that is monitored is:

- Processor utilization
- Memory utilization
- CPU usage per task
- Response time per task for users
- Printer management - the number of prints, queue length, time for print outs
- Storage capacity
- Number of clients
- Number of logins
- Number of simultaneous users

In Debian Edu, Nagios is used as a monitoring tool.

4.3.2 Analysis

On the basis of data collected from monitoring routines, one tries to identify any bottlenecks in the systems. Examples:

- Poor or varying utilization of the hardware
- Poorly designed software
- Poor utilization of memory capacity
- Bottlenecks on data storage, memory or processor
- Bottlenecks in the network

4.3.3 Configuration

If the data analysis uncovers bottlenecks, one needs to try to set up the system in a way that better caters for the users' needs.

Here is a list of commonly encountered bottlenecks and what to do to get rid of them:

Bottlenecks	Actions
Missing sound, USB stick support and DVD on thin clients.	Install diskless workstations (> 800 MHz processor, > 256 MB RAM)
Has 60 thin clients connected to the server and wants more PCs.	Go for diskless clients, or install another thin client server
Thin clients run slowly after we expanded with 20 pieces without acquiring a new server machine	Install 2GB more memory on the server machine
Thin clients with 32MB memory do not start after upgrading to Skolelinux 2.0	Turn on swapping on the thin clients, or downgrade to LTSP 4.2 which is set up with swap.
Flash animations make the thin clients slow when 50 students are logged into the same server machine	Install diskless clients

4.3.4 Implementation

Implementation of possible changes to the system configuration must be done in accordance with the guidelines set for changes of the system. A well-planned function and performance test must also be done before changes can be made in the production system. Testing is done to avoid operational disturbances when changes are set into production.

4.3.5 Preparing the capacity plan

A capacity plan is basically an investment plan for the ICT system based on knowledge of the users' current needs and future plans.

The capacity plan should be updated and processed once a year, normally in conjunction with the budget process. The plan should include the following themes:

- Introduction
- Preconditions
- Summary
- Current and future user needs
- Service summary
- Resource summary
- Areas for improvement
- Cost model
- Recommendation

4.4 Availability management

Good and stable availability of ICT services is obviously crucial for users.

Availability, seen from the user perspective, depends on the following assumptions:

- Availability of technical components

- Failure tolerance
- Quality of maintenance and support
- Procedures and routines for processing operational services
- Security, integrity and availability of data

Availability can be measured in several ways. But before we show examples we'll point out what may be difficult targeting figures. If we should make systematic efforts to availability, we have to clarify what the different things mean. What means for example a percentage of availability.

Let's say a "computer with computer program" is a service. If the computer program does not work one day, then the service is unavailable if all the other programs work fine. What if the computer program is unavailable for a classroom, but available for the rest of the school (because of an underlying service). This is a difficult matter to clarify and work on in practice.

4.4.1 Availability measurements

Availability can be measured using several methods. Here are some examples:

Value	Meaning
% available	The value can be availability between hours 08:00 and 18:00. If the system is down 1 hour during one day, than the system is available in 90% of the agreed upon time. If availability is measured over a month with 20 work days, then the system is available 95% of the time.
% unavailable	Is the system down one hour during an agreed uptime, for example 10 hours a day, the system is unavailable in 10% of the time. Measured over 20 days, we may assume the system has been unavailable for 5% of the time.
Downtime	One can agree on the number of times one accepts the system to be unavailable during, for example, one month (20 days). It can be a maximum of one hour unavailability in that period, and between 08:00 until 18:00.
Error frequency	Even error frequency can be measured per day or for each month. 3 errors in the month because the system was down between 08:00 and 18:00, is an example.
Error consequences	Measured values are a common starting point for judging how to respond to an error beyond ordinary error correction. The customer or the school for example, may ask to pay less for the operating agreement for the current month.

The most important is that your measurements describe the user experience in the best possible way. Therefore, one should measure what is important for the user.

The feedback from the schools is that printers give most problems. This includes everything from the print queue has stopped, to missing paper or toner. Some have also experienced some instability with the browser, and that OpenOffice.org suite is hanging. It may happen when your broadband connection is unstable and you have links in documents going to the Internet.

4.4.2 Infrastructure

To have a stable computing environment, one is dependent on a good enough technical quality of the network. Several schools have experienced instability because the physical computer network is provisional and of poor quality.

Today many invest in wireless networks. Doing so, one must also be aware of wireless networks having significant weaknesses. Wireless networks have limited capacity. It can be quite choppy when about 30 students are to see a film from the Internet simultaneously. Wireless networks also have shadows. Meaning areas may not get coverage, which allows some to end up in blind zones. This would provide poor or no net connection at all.

Availability requirements for the maintenance company and ICT service providers should specify good quality of network services to schools.

4.4.3 «Single points of failure»

Some parts of the system simply must work. Failures in a firewall, for example, may compromise security or (if you're lucky) shut down the whole network. This last can also result from problems with the DHCP (Dynamic Host Configuration Protocol) system for sharing out addresses.

The operating department has a responsibility to know which parts may stop the entire system. It is important to find these points, and remove the errors one by one, to the extent you can afford. If one can't afford to remove these sources of errors, one must live with the risk of the entire computer network suddenly grinding to a halt.

Sources of errors making everything stop, may also be logical rather than physical. This is especially true for computer networks and databases. So it is important to have a broader perspective when it comes to such errors.

4.4.4 Risk management

One must consider what risks one accepts in the network. Is it acceptable that users lose personal files and data, when a hard drive fails? How quickly should one replace broken equipment? Some schools have spent several days getting a server up and running again after a virus attack. The municipality may have no resources to allocate to fix errors.

Much of the operations work goes on to maintain the agreed service level. It's about avoiding and losing confidence and user satisfaction. Risk management is about having in place the appropriate resources to keep the entire computer system on the air, and have resources ready if something should go wrong, and needs to be fixed.

4.4.5 Testing

It is a big difference to install equipment and software on a single PC and to do it on hundreds, even thousands of computers. With the responsibility for hundreds of machines, a small error that one can live with on a PC, means much instability and discontent if it affects hundreds of users.

To avoid making mistakes during installation and to contribute to stability, it is essential to test the equipment and software to be used. It's about following up the expected quality. If you want stable operations one must often choose next to the last edition of equipment and software.

One should avoid adopting software with a version number ending with a zero. For example you should avoid OpenOffice.org 4.0. One should adopt the office program when version 4.0.2 has arrived or later. Then the program has been fixed for several errors. The same applies to hardware.

Server machines have usually a slightly older version of processors, and more robust memory, and hard drives. This is because many people use this hardware simultaneously. A small error that would not mean anything for one user, can provide downtime if 30 users are logged into the machine.

So testing is about to use proven equipment and editions of software running well a half or a year. Testing is also about trying out the different parts in a smaller but realistic context, to ensure that everything works. Adopting the latest version, or even beta versions of software or completely latest hardware usually lead to much trouble and extra work with maintenance. Setting systems in production without a small test in realistic environments usually lead to significant firefighting and dissatisfied users.

When testing in a smaller scale on equipment in production, it is essential to coordinate that with those affected. In addition, one must choose when to test. One should not test new things, for example, under ongoing exams, with the use of ICT tools.

4.4.6 Design improvements

It is often worth an operations department's while to enhance systems that produce many operational messages. If users get much spam, then it might be wise to install spam-filters. There might be a lot of extra work with students who constantly forget their passwords, if teachers have to get central sysadmin staff to help them out. To avoid extra emailing and double work, the teacher can be authorised to give the student a new password.

These are two examples of design improvements that simplified maintenance and made users happier. A well-run maintenance team has a prioritised list of such improvements. Prioritising these, as a rule, is based on how often relevant issues show up in the service office's message log and estimates of how much work each improvement shall involve.

4.4.7 Planning for availability

It means having realistic expectations to the ICT service based on what operations costs. Plan for what's the expected accessibility. For example, when schools require one should be up and running in less than 1 hour after the server crashes, one must have a standing pre-installed machine in reserve, to be inserted as a replacement for the faulty machine. What should be done during one hour is to copy your backup files to the backup machine.

For when a diskless or thin client fails, the school should have a small store of machines and monitors prepared. The school ICT contact can fetch and install a replacement machine. This can be done easily without waiting days for an equipment order to be filled.

4.4.8 Planning for recovery

As for equipment standing ready to replace any that develop defects, users also expect to be able to retrieve lost files and data. Therefore it is crucial to back up user data regularly and keep a copy of the configuration files. One must also have architectural diagrams, and descriptions of systems, to enable ICT staff to quickly set systems up when something goes wrong.

It is crucial to schedule backup of user data and settings. One must plan ahead in order to have proper equipment and appropriate services. Routines must be planned to be followed when certain error situations occur and systems must be restored.

4.5 Service Continuity

Operating continuity or continuity management is often the most costly part of the work. High demands to operational continuity will require huge investments, which must be agreed upon whilst making the SLA. For example it can be agreed that there is no disaster plan for certain services. If you have a disaster plan the value is very low if not tested once in a while. Usually this is expensive. There are examples where customers and management have blocked the engine room and turned off power to test readiness of the IT department.

Operating continuity may be more needed in certain periods like under the examination periods. Then extra requirements can be needed in order to have equipments with backup ready in case of a hard disk failure on the server. But even this will require considerable additional work for the operational staff.

An IT coordinator told us that it might be just as well to postpone the exam one day, if something went wrong with the computer system. This costs a lot less than having a double number of servers at each school. There are examples of schools having had water leakage. Then it is usual to defer examination a day or two to repair the damage . One might think the same way when it comes to school data solutions. If you have a backup of home directories for pupils and teachers, you have time to consider without doubling the systems at each school. Then it is sufficient with one or two servers in reserve located at the municipality building, which quickly can be moved and connected at the school if something goes wrong.

5 ICT Infrastructure Management

This part of the operational documentation is to a greater extent about technology. The other chapters about service support and service delivery, are about work processes and procedures. Infrastructure management, are about planning,

design, deployment, and ongoing maintenance of ICT systems. The purpose is to provide ICT solutions adapted to the organisation's needs, which can be operated over time to a cost one can afford.

Good planning, administration, and management are the key to ensure a well-developed ICT service, and that the service can be adapted to changing organisational needs over time. It is about using resources well and having skills and competences required to offer a good ICT service.

Even with a well-developed infrastructure, one must expect 60-70% of the cost to go on operations, i.e. service support and service delivery. Likewise, infrastructure constitutes around 20-30% of the total costs, and one must take this part just as seriously as operations. The infrastructure chosen also has a major impact on what operations will cost and what the systems can deliver.

Most people associate infrastructure with roads, water and sewage, and power supply. Building a house requires infrastructure in place if we are to have a certain housing standard. In the computer world infrastructure is often associated with the data network. This was in the 1980s. Over the next two decades infrastructure extended to networks, computers, software and maintenance. So in this part of the documentation is all network, hardware and significant portions of the software part of the infrastructure.

Even here we focus on practical planning and implementation. We have gathered concrete planning data from different municipalities with good ICT plans by budgetting and procurement. We went through design and planning, deployment process, operations process and support. It is important to keep in mind the difference in operational terms of what Service Desk does with, for example. support, and the support operations does for example with network cables to the school. It is basically four processes that recur at infrastructure management:

<ol style="list-style-type: decimal;"> Design and planning process Development and maintenance of ICT strategies and processes for deployment and implementation of appropriate solutions in the ICT infrastructure of the organisation.   Deployment process Concerns implementation and deployment of activities and / or ICT solutions designed and planned with a minimum of disruption to the organisation   Operating process All activities and initiatives to deliver and/or maintain the desired use of ICT infrastructure.   Technical support process The development of knowledge for evaluation, support and quality assurance of all current and future infrastructure solutions.

5.1 Design and planning

Design and planning is about providing pervasive strategic guidelines for the development and installation of an ICT infrastructure to the institution's needs. It applies not only to infrastructure networks, computers, and applications. Also basic processes must be in place to get the technology to work. It applies to both Service Desk and processes for service management.

Avoiding scheduling, or cutting corners, is very risky. Therefore it is often wise to spend a little more time and effort on planning, which will reduce risk and provide significant benefits during implementation. Most projects that get stranded do so for lack of planning. Putting in place an ITIL process in an organisation depends entirely on preparation and planning, along with effective use of people, processes, and products (tools and technology).

It is very important to communicate with and talk to every part of the organisation while planning ITIL. In Norway this is regulated by the labour act §4-2:

- *Employees and their representatives should be kept informed of systems used in the planning and execution of the work. They should be given the necessary training to be familiar with these systems, and take part in designing them.*

The objective is to deliver the right ICT solutions for your organisation. These must be easy to maintain and adapted to the school's needs. The solution must be reasonable over a long time, also when the systems expand. During a design and planning process should one relate generally to a steering group and a reference group. A good project ensures to have skilled people in the steering group and people who contribute to the reference group. A good planner is clever to use those groups and other employees to bring up the good solutions.

We created a check list over activities and deliveries in an infrastructure project.

Feedback

- Plan for the schools, both curricula and activity plans
- Existing ICT strategies
- What's expected of operational services
- Current ICT systems and operations management

Processes

- go through all the suggestions and documents
- look at others performing design- and planning activities
- making and maintaining ICT plans and decisions
- making and maintaining ICT architecture
- making and maintaining the ICT strategy

Deliveries

- ICT-strategy
- ICT decisions (with justifications)
- IT plans
- the entire IT architecture
- Design and planning of processes and procedures
- organisational structure and framework
- Design, planning standards and decisions
- SWOT analysis (Strengths, Weaknesses, Opportunities, Threats)
- user cases and usability studies
- Requirement lists and tender documents
- Project plans
- Technical drawings, plans and maps
- Comments and feedback

As you can see extensive planning is carried out for an infrastructure project. An ICT project for schools in the municipality can quickly reach several million NOK (hundreds of thousands pounds) should one deliver 500-1000 computers with power, computer networks and software. With such amounts, it is important to have good and feasible plans with realistic budgets.

There are several examples of municipalities that have underestimated the need for funding for ICT in schools. They have installed lots of nice equipments remaining unused. Maybe the software could be missing. The network might be of poor quality or lacking power connectors. The municipality gets quickly an additional expense of 2 million NOK (around 165 000 pounds) if 800 machines in 10 schools should have a computer network and power connectors.

Good development plans are made to avoid surprises. Plans are also made to ensure a proper level of ambition with a realistic budget.

5.2 Deployment

Definition:

- Is about implementation and deployment of activities and/or ICT solutions designed and planned with a minimum of disturbance for the organisation.

The planning project shall have assessed what equipment schools have, and how much is available. From that, one creates a plan to roll out new equipment, or replace equipment, at each school and at the central operating service.

This is about placing the equipment where it is to be used. Set each PC on a table and connect it to the computer network with a network cable. Plug the power cable in to the electrical outlet. Connect the screen and put network cables in the correct switches.

The term roll-out (or deployment) is used both for placing the equipment, and for installing and configuring software on many machines. Rolling out software may also be referred to as "release management". But the word roll-out is short and fine, although one should make clear whether you are talking about hardware or software, which require totally different procedures.

Rollout management is about implementing what has been planned and designed in the release process. Getting out the equipment where it should be is often harder than we think, and takes considerable time. This is because many parties are involved either to deliver the equipment, or being among the many to receive it. In a way you could say that rollout is the same as a wheel bolt that holds the vehicle wheel in place on the shaft.

To get everything in place, is totally dependent on a lot of coordination. One must make good tactical planning, which involves both change management and project management. One must make sure that the roll out is associated with the design and planning process.

Danger may often occur by underestimating how deployment affects existing systems. Taking in use new solutions or upgrades will affect or change the organisation. Work routines are changed and one gets new ways to solve tasks.

In the school , the change means introducing ICT tools in school subjects. This is new and different for teachers. Many are unfamiliar with how the equipment can be used in teaching. At the same time, an operating and maintenance service should be in place to give schools a safe and stable ICT solution. This leads to changes in the organisation, which must be planned and requires resources. Therefore it is important to take this into consideration both under planning and rollout..

5.2.1 Roles during roll-out

Building an IT infrastructure can be likened to building a house. When building a house one really needs an architect, a builder, an owner, masons, carpenters, plumbers, electricians and one or more supervisors (foremen). Much the same applies to deploying infrastructure. We have summed up the roles recommended as part of the operating standards ITIL.

- Owner of the deployment process - is responsible for the deployment process, and for it happening in a good and efficient manner.
- Project manager for the roll-out - is responsible for developing appropriate plans for deployment of the ICT system and day-to-day directing of the roll-out.
- Coordinator for the roll-out - is responsible for coordinating roll-out activities. The Coordinator shall ensure that the project attains the objectives and acceptance requirements for the system and ensure an orderly handover.
- Roll-out analyst - responsible for ensuring that equipment is installed in suitable settings. Shall verify that the equipment and premises are suitable for the standards, tests and deployment agreed upon..
- Employees in the rollout team - are responsible for the ICT solution and the work environment, and support the acceptance and test processes.

As we see, roll-out touches many parts of an organisation. Technically it affects configurations and versions of software and equipment. It also influences the process of change and how work is done at the service.

One must think carefully about who gets assigned each of these roles. Even in a full-scale roll-out project costing several hundred thousand pounds, one person might have several roles. But it is not always prudent to give one person several roles, because it can be very demanding to follow up with both suppliers and customers of the equipment.

For minor upgrades and adjustments, there might be soon too many roles. For example, one needs not to have a project manager to place a new server, or replace a switch. This is part of the infrastructure, but is very close to the operations and maintenance. The important thing here is to distinguish between a rollout in infrastructure, and operational services. The operations department will not take over the equipment before it operates as agreed. In other words, you have a transfer document where one acknowledges that the equipment is delivered as agreed.

5.3 Operations

Definition:

- The development of knowledge for evaluation, support and quality assurance of all current and future infrastructure solutions.

Operations of the equipment is about having tools and machines in place as the basis for delivering the agreed upon ICT services. Operations of equipment has a strong focus on technology. This supports all other activities being done with the ICT systems. Often operations are seen as support tucked away in an office at the end of the corridor. As a hygiene service" .First when something goes wrong, operations personnel are contacted. A good operational service is still crucial for ICT tools to work properly. Without good operations one accepts the loss of time and that tasks cannot be solved. For example a school can get problems when using tests done with ICT tools.

One may ask whether one needs operations. Needing people to operate in today's high tech world? Is there no one who has found a way to solve operational tasks automatically? Why should you have people in the operations? The answer is usually that balancing between what is done automatically, and what people have to follow up with. An important realization is that most people want someone to talk to when a problem occurs. They want the error to be fixed, and they want feedback that everything goes smoothly. This type of error correction is not particularly easy to replace with machines.

A good operations department chooses to automate where possible. Simultaneously one needs people to monitor and keep control of the automated solutions. The automation must be further developed. There are also situations where automation is insufficient. Equipment breaks, and applications crash. You need someone who is handy and can rectify errors, or obtain substitutes for what cannot be repaired.

A poorly organised operations service wastes a lot of time fighting fires and working through manual procedures which could have been avoided with automation. Time spent automating can quickly pay for itself by freeing up time. This time can be used to improve support, provide more services and raise the quality for users. To get a permanent fix of errors, sometimes people might have to delay upgrades or remove services for temporary repair. This makes time to fix problems properly when monitoring the system manually would otherwise take up all the time.

Operations is mostly about preventing errors, or correcting equipments with reported errors. One is often not entirely familiar with causes of an error. One must troubleshoot to find the fault. Good operations' employees have flair. They use past experience to uncover errors. Then they go almost right to problem solution and correct the error.

5.4 Configuration item

A Configuration Item (CI) is part of an infrastructure. It often describes a wish for change or a question. Maybe to put in place a new service, or make adjustments to the services you already have in production. Often it is a question of upgrading some equipment, or obtaining something new.

Configuration items are important in configuration management when it comes to equipment and infrastructure. Often a configuration item is about whether systems shall:

- be run

- be closed
- be shut down
- be started
- be interrupted
- be removed

5.5 Technical support

Technical support ensures staff are available with the right skills to support services provided in the computer network, as well as staff to work at the Service Desk. As part of technical support one should have in-depth documentation with technical advice. The advice should provide information, guidance and examples of roll-out activities, and describe support and maintenance of all parts of the ICT service. To achieve this, the staff must know, or be able to obtain information, about the technologies, processes and documentation in use. As listed below, technical support consists of several activities:

- Research and development related to new technology.
- Third line service in response to incident reports from the Service Desk and general handling of problems.
- Delivery management - technical support lack in-depth knowledge or understanding of the technology in use and need technical support from others.
- Coherence of design and planning. Especially in support and documentation. For example, when preparing tender documents.
- Coherence with the deployment of new system versions, and acceptance in the operating environment.
- Analysis, interpretation and distribution of information from reports and logs.
- Tactical assembly of improvements in the quality of the ICT service delivered.

6 A design and planning example

As an example of how infrastructure can be made, we have taken significant portions of the ICT plan for schools in Nittedal 2005-2008. We have made some adjustments, to make it more general and easier for others to copy.

- Background for the plan
- What's expected of the ICT tools and services
- Skills needs
- Investments
- Objectives
- Students and teachers
- Status and objectives
- Costs
- Other purchasing options
- Software, learning platforms, and services
- Software and learning platforms

- Online services
- Resource usage
- Centralized operations and roles
- Operation and support costs
- Recommendation
- Attachment

6.1 Background for the plan

In its "Program for Digital Competence 2004-2008", the Ministry of Education and Research sets objectives for the use of digital technology in the Norwegian school. "By 2008 we will have an infrastructure, an organisation and a culture that makes our school system to one of the world leaders when it comes to development and educational use of ICT in teaching and learning."

Being able to use digital tools is defined as a basic skill for all 13 school years. Pupils' development of basic skills is to be given priority in all subjects. The new curricula will mean that students must increasingly make use of digital tools when learning. Pupils should be able to use the same technology in work, that forms the basis for the final assessment, as they use when learning. When the examination is carried out using digital tools, this provides better coherence between the learning process and the final assessment.

A national survey in Norway (Skolenes digitale tilstand 2003, ITU, Feb. 2004) shows computers are seldom included in the subjects in primary school and computers are barely used by pupils in the school.

This plan builds on "Competence Plan for schools in Nittedal (2005-2008)" and implements that competence plan's objectives for digital knowledge in Nittedal's schools. In addition, this is a plan for investment and resource requirements associated with the operation of our Linux network.

6.2 What's expected of the ICT tools and services

We have different objectives for different groups in the school and to different aspects of the ICT commitments. Briefly put, our goals are:

- Get increased use of ICT among both students and teachers by increasing physical access to ICT equipment.
- Be tools-oriented, and therefore emphasise the use of ICT tools in the school's subjects.
- Give full access to educational software for everything from musical composition and use of the Internet to learning to write, simulations and games.
- Be thrifty and utilise the financial resources we have in the best possible way.

Through these main objectives we will achieve:

- Teachers get good working and communication tools at work.
- Students get the opportunity to be personal users of ICT and use ICT as a natural tool in everyday school life.
- The school will be physically able to fulfill various aspects of the curriculum related to ICT.
- Operating and maintenance costs are not greater than what the school budget can allow.

6.3 Skills needs

To build and maintain the infrastructure you need a collaboration between many different professionals. As an example, we show what equipment areas you need expertise in. These are equipment areas that form part of the infrastructure in an ordinary school.

- Network infrastructure with local area network (LAN) and wide area network (WAN). Mostly it is easy to obtain switches and other network equipment. This is off-the-shelf kit. But it must be set up for the planned architecture designed for centralized operations. This is a job for professionals. The municipality building department must approve the changes made.
- Power supply (230V/110V) supporting client computers, servers and network equipment. Many schools have not installed outlets for all computers to be placed in classrooms, computer labs or the library. It is a job for professionals to plan a power grid with enough sockets, and follow given regulations. The municipality building department must approve the changes made.
- Servers and clients support a greater variety of network services and end-user applications. To obtain the right equipment is a considerable job. It is about finding equipment with appropriate capacity, good quality, decent guarantee schemes, and low prices.
- Machine setup and systems for monitoring hardware. To be sure that all equipment is running, it is normally accompanied by remote monitoring systems. That way you can have an overview of the health status of the equipment in a centralized operations center.
- Designing appropriate environment or room for placement of equipment that needs cooling. Computers and network electronics emit considerable heat. First recently, manufacturers of equipment have addressed the ever-increasing power consumption. Therefore, one must sometimes ensure that the excess heat gets transported away. Such cooling systems need to be installed by professionals.
- Knowledge of different performance requirements for software. A program for video editing must run on a workstation with $>$1.5 Ghz processor and ample memory. Other programs may easily be used on a thin client. One must have relatively good knowledge of what can be expected of different types of client machines to choose the right mix of equipment. This requires insight into how computers are intended to be used in different subjects and in different rooms of the school.
- Installation and setup of optional equipment such as printers, video projectors, computer boards and the like. To set up accessories may quickly take considerable time. For example video projectors need to be screwed into the ceiling, and one must arrange for both screen cables and power to reach them. Printers must have a network point and be connected to the network. This type of installation usually requires professionals for both installation and setup.

Besides to the different professionals needed to build the infrastructure, you will additionally need:

- Owner of the deployment process - is responsible for the deployment process, and for it happening in a good and efficient manner. This may be the steering group.
- Project manager for the roll-out - is responsible for developing appropriate plans for deployment of the ICT system and day-to-day directing of the roll-out.
- Coordinator for the roll-out - responsible for coordinating deployment activities. Coordinator shall ensure the project fulfils the objectives and acceptance requirements that apply to the solution, and ensure an orderly handover. This may be an assistant to the project leader.
- Deployment analyst - responsible for ensuring an appropriate environment at the locations where the equipment will stand. Shall verify that the equipment and premises are suitable for the agreed standards, tests and deployment. This may be an assistant to the project leader, with the task of reporting deviations from plans to the steering group.
- Employees in the roll-out team - responsible for ICT solution and the working environment, and support for acceptance and test processes. This is employees who participate in one or more sub-projects.

Organizationally it will look like this

Organisational part	Tasks
Reference group	shall represent users of the system. They will advise on measures to promote a good and everyday ICT solution for schools.
Steering group	whose mission is to ensure the project has enough resources and that project management gets the roll-out carried out according to the plans. The group will consist of skilled professionals who are well acquainted with project implementation, system solutions, and the use of ICT tools in schools.
The project	has the task of building the solution. The project usually consists of many sub-projects, which deliver their part of the solution.

6.4 Investments

To meet a new curriculum, schools must have sufficient computers available for their students and staff. This investment plan includes the actual costs of increasing the stock of computers at schools to reach national objectives. Those objectives call for at least one client machine or PC per four students. The requirements for equipment shall probably increase in a few years, so we expand that to a PC workstation per three students. All teachers should have access to a computer in their daily work at school.

Today the school net consists of servers and thin clients at the schools, and a shared server for backup in the municipality. Since we can use used computers as client machines in our network, the users' computers are not the most expensive (we buy used equipment and receive donated equipment from industry). The major costs lie in increased need for power outlets in classrooms, and possibly an increase in electricity bills in the schools.

The increased number of concurrent users will also increase support and operational costs. There will also be a need for tables and chairs for the new PC workplaces. In addition, all schools received a broadband connection at a fixed price. Later, we shall illustrate the total cost by doubling the stock of equipment.

Status for pc coverage 01.06.2005 is:

1. 8.9 students per computer in primary schools
2. 4.4 students per computer in secondary schools.

Objectives for the students:	Each student group (formerly called classes) should have access to at least five computers, plus the school should have a computer room with a minimum of 15 PCs. In addition the school needs some special equipment for video editing, special education and reading/writing courses.
Objectives for the teachers:	All teachers should have access to a computer in their daily work at school.

Total number of machines:

Status per 01.06.05		Needs 2008				
Server status		Clients	Laptops	File servers + thin client server:	Clients	Laptops
Holumskogen	1	25	5	2	68	15
Ulverud	1	35	5	2	111	15

Slattum	1	44	8	2	87	15
Rotnes	1	35	5	2	80	15
Sørli	1	31	5	2	60	15
Kirkeby	1	31	5	2	94	15
Hagen	1	7	5	1	46	15
Li	2	70	5	2	130	30
Nittedal	1	55	20	2	110	30
Hakadal	1	45	5	2	52	30
Sum	11	378	68	29	838	195

We envisage a combination of thin clients, diskless clients and laptops. Schools should have an infrastructure making it possible to roll out thin clients in every classroom. Here students can write, calculate, using the internet and make presentations. In addition the school will have the opportunity to lend laptops to different groups. In this way students get close to full PC coverage in certain work situations. The laptops are connected to servers in wireless networks. In this way teaching becomes more flexible.

6.4.1 Pupils

We recommend an investment providing at least one client computer per third student, something the government has stated as a goal for ICT tools in schools. To achieve this we need nearly a doubling of the number of client machines.

6.4.1.1 Status and objectives

To reach our target we need to increase the machine park from 506 to 1033 machines. This is an increase of just under 600 machines. (thin clients, diskless clients and notebooks).

6.4.1.2 Costs

We have counted with these prices, which might be subject to changes:

- Thin Client: 700 NOK, per piece
- Server: approx. 50,000, per piece
- Monitors: 500, per piece
- Portables: 8,000, per piece
- Power plug 750, per piece
- Table/chair: 700, -
- Increased resource here means increased number of hours for ICT contacts in schools. Here we consider an hourly rate per teacher of NOK 270 (£22.5) - per hour, or NOK 467,100 (£38,925) a year. We also consider somewhat increased resources to central operations of the municipality. We expect slightly less than one new hire in centralized operations for over 1,000 client machines. In addition, ICT contact at each school, training and ICT coordinator.
- Licence costs. Today, we can install Linux on laptops using the school's existing network. Thus we avoid the rent of Microsoft products such as Windows and Office. School Rates for rental of Microsoft program cost as much as all computers over a period of 5-6 years.
- Broadband agreement, all schools have broadband connection. Price depends on the individual school agreement.

Recently used equipment have more performance than the machines that were available for 3-4 years ago. If the users machines had 256 MB of memory and 800 MHz processor then those would fit as diskless clients. This simplifies support for using the CD / DVD player, audio, USB stick and similar.

2006	**2007**	**2008**	**Total**	
Thin clients and diskless workstations	130,000	130,000	130,000	322,000
Servers	500,000		500,000	1,000,000
Monitors	80,000	80,000	80,000	230,000
Laptops	340,000	340,000	340,000	1,020,000
Other: switches, cables,	150,000	150,000	150,000	450,000
Power plug/cables	290,000	290,000	290,000	870,000
Tables/chairs	190,000	190,000	190,000	570,000
Increased resources, operations	700,000			
Licence costs for portable machines	40,000	40,000	40,000	120,000
Broadband agreements	100,000	100,000	100,000	300,000
Sum	5,582,000			

6.4.1.3 Other purchasing options

There has been a growing interest from politicians, parents and teachers to go over to laptops in secondary schools. Laptops and a wireless network will give schools a completely different flexibility in room layout and teaching.

The problem with only focusing on laptops is:

- We have to buy Microsoft licenses in addition to the machines.
- The machines have a lifespan of approximately 3 years. Thus the municipality incurs an annual expenditure to cover new classes in secondary schools.
- Increased insurance costs
- A greater need for power outlets when all laptops must have access to electricity.
- Increased need for the schools ICT-contact resources
- Doubling of central operating costs of preparing the disk images etc. for laptops, and maintenance of a locally-installed system on 266 additional laptops.

This option has a total price tag of roughly 12 million NOK, just over 1 million pounds. (This does not include any increase in insurance costs.)

6.4.2 Teachers

Each teacher should have access to a client machine at the school.

6.4.2.1 Status and objectives

Status: Schools today have approximately 65 PCs divided among approximately 266 employees. This gives a PC coverage of 4 teachers per PC.

We want to give credits to the teachers in Nittedal. The new curriculum sets high requirements for teachers' ICT skills. It will be necessary to ensure that all teachers in Nittedal have access to a computer. Today's teacher plans and implements

teaching on and with data. They document and report, write weekly plans, work plans, annual plans and IEPs. More and more teachers use e-mail for contact both at home and school.

Schools have already arranged to buy computers for their staff. As a result, the number of computers varies from school to school. We aim to give each teacher access to a computer at work.

Here we outline two options to achieve full PC coverage for teachers in Nittedal.

6.4.2.2 Costs

Option 1: Thin clients in combination with portables. This will give each teacher access to a thin client + that 3.3 teachers share access to one laptop.

cost	Total cost		
Alternative 1	Thin clients	140,000	
Monitors	100,000		
Laptops	640,000		
Other: switches, cables,	80,000		
Tables, chairs	140,000		
Power plug/cables	400,000		
License costs	40,000		
Total	1,540,000		
Additional for flatscreen	200,000		
Totally with LCD	1,740,000		

The advantage of the thin clients to teachers are the low cost of procurement. We can also expect a longer lifetime of the thin clients compared to laptops.

But reused equipment is often without flatscreen. Client machines' cabinets can be large, giving space shortage in the workplace for teachers. Should we obtain a flatscreen to all teachers, one must triple the cost, going from 500, - NOK (43 £) to 1500, - NOK. The total cost would increase by 200,000 NOK. Overall equipment to teachers cost 1.74 million NOK (150,000 £).

6.4.2.3 Other purchasing options

Option 2: Portables for all teachers

cost	Total cost		
Alternative 2	Laptops	2,128,000	
Wireless switches	80,000		
Power plug	75,000		
License costs	117,000		
2,400,000			

The problem we see is the actual location of the thin clients. Teachers have small work desks often in large common rooms. A thin client per teacher with old-style (CRT) monitors would create a space problem in all schools. The problem is greatly reduced with modern flat screens.

The advantage of portables is that they require little space. Teachers can easily bring their work home. The disadvantage is the lifespan of a portable, around half that of stationary equipment. It is reasonable to assume that laptops are twice as expensive to maintain as desktop PCs, and three to four times more costly to operate than the thin or diskless clients.

6.4.3 Recommended technical development budget

In the period 2005 to 2008 we put up the following recommendation for an IT infrastructure at the schools.

Number	Article	Cost
600	Thin clients and diskless workstations including all infrastructure	5,582,000
200	Thin clients or diskless workstations with flat screen and all infrastructure	1,740,000
800 client machines in total	**Total**	**7,322,000**

6.4.4 Software, learning platforms, and services

Where software runs, depends on infrastructure and capacity of the network. It's fine to operate all installations on schools from a central location, for example from ICT services in the municipality or a centrally located operation.

One must take into account that network capacity to schools can provide limits to how much schools can download at the same time, or where it is best to place servers to get the equipment's full functionality. There is a big difference between a single teacher downloading a film from for example NRK, compared to about 30 students doing the same simultaneously. If the shool has 1.5 Mbit/s broadband capacity, it is not possible for 30 simultaneous users to download the movie directly from NRK. Then you must have cacheing proxies in place at the school.

6.4.5 Check list centralisation

UNINETT ABC has made a document with recommendations <<FootNote(Recommendations of UNINETT ABC: http://www.uninettabc.no/?p=veiledning&sub=annet)>> related to the centralisation of ICT operations. It gives advice about the placement of servers and which operational tasks may be centralised from the available capacity of the bandwidth to the school.

General measures to improve the operation of clients and servers	**Thin or diskless clients against local serversLockdown of thick clientsLocal server machines**	**Remote operationCentralisation of some functionsLocal server machines**	**Server machines regionally/nationallyCentralisation of all operations**
Capacity of the schools' network	Low bandwidth (ISDN)	Medium bandwidth (ADSL and similar)	High bandwidth (fibre and similar)

6.4.6 Software

The new curriculum (L2006) highlights the use of digital tools as one of the "basic skills". We wish that the use of ICT should go beyond teaching, and increasingly provide educational and administrative tools to support learning activity and new forms of learning while providing easy access to knowledge. Giving experience with digital learning platforms is one of the objectives of the competence plan. The aim was for one or more schools to try this out in 2006.

Research shows computer equipment being used to a limited extent for teaching in schools. Computer usage has stagnated and in some subjects declined, research shows (ITU Monitor 2005). The use of ICT in schools is often individualized, and students learn to become consumers. Teaching methods are hindering the sharing of knowledge in school. Few teachers use ICT daily. Internet and text-related services are the most important forms of computer use in schools.

Simply put, teachers focus too much on using tools for office administrative work such as MS Office or OpenOffice.org. What they ought to focus on was the use of simulations, editing pictures, audio and video communication on the Internet, and games.

Home use is often quite different. At home students are producers and use ICT mostly collectively and for communication. They put together and send each other pictures, exchanging content, using the major opportunities for recording, editing

and sharing of movies which is possible with today's computers with broadband. Children and young people also play video games more at home than at school (ITU Monitor 2005).

Researchers say that video games are one of the main leisure activities for children and young people. One child in four plays every day (Youth Agency 2006). Video games are a social activity. In the wake of the games both virtual and physical communities arise, from playing together on consoles to participating in gatherings where youth can play.

An important task is that the school development contributes to update general education perspective in the general part of the curriculum. Allowing digital judgement or digital education developed in relation to steps and learning strategies, as stated by the national Research and Resource Centre for ICT in education.

To get equipment more in use requires considerable effort by the teacher. They must be continually educated in new forms of learning to use the new ICT tools for teaching. There must be more emphasis on youth's actual use of media and forms of communication. It is not enough to provide a teaching platform and e-mail. The tools should fully support the new ways of using media.

To achieve this, the equipment must be adapted to the software and the online services that teachers and students use in their school work. The browser is arguably the most important program students use in learning. Many will also be surprised that office programs as OpenOffice.org or MS Office are irrelevant in lower grades. At those grades, simple programs for practicing writings, drawings, communication, simulations and music forming apply. So what is important in the choice of software is to provide good access to the Internet and support for active learning using ICT tools that are relevant to the school subjects.

With diskless clients you get full support for multimedia, film, USB sticks and more. The advantage of thin clients is that they allow reuse from as far back as 1995. At that time the machines had no video capabilities. The USB standard was not fully developed. Computers from 2000 and later usually have a much higher capacity. Such machines can easily show video clips from NRK, DVDs, and one can play games.

The advantage of diskless clients is that they provide the same performance as the so-called thick clients or computers with most of the software installed locally. At the same time one gets the same low operating costs with diskless clients as with thin clients. This is because all the software is managed on the central server machine.

Today Skolelinux comes with over 50 school-relevant programs. In addition, it has browser, email client and OpenOffice.org with 8 different office applications. This is much more than what comes with Microsoft which mostly offers browser, e-mail and 5 current office tools.

With Debian Edu, it is also relatively easy to customize menus for the various stages of education so that we can reduce the number of educational programs. Especially since some applications are introduced in 4th-5th grade. While programs may be popular at first or second stages of education will be too easy when students have gotten older and learned more. In addition, there is an increasing number of educational programs on the Internet. This is software that works on any platform. So you can use the programs at home on Apple or Windows, and the school's Debian Edu. Students handle this just fine.

6.4.7 Learning platforms

Various digital learning platforms are found on the market. Some cost money, others are free. They all offer teachers and students an area where they can share and store documents, send and receive information.

Product		Price example:
Digital learning platform	It`s learning	3300,- NOK per school per year
School network		Free

6.4.8 Online services

Regardless of the bandwidth the following functions can be centralized:

- Configuration management, i.e. having oversight and control of the configuration of machines, networks, applications and services
- Program management, i.e. having oversight and control of access to, usage and performance of applications and services

- Updates and patching
- User administration, preferably with a FEIDE compatible user management system (BAS)
- Licence administration
- Monitoring and measurement

Checklist for services that can be centralized or replicated. For example can backup be centralized. The same applies to the user database with a central directory server (LDAP) with replication to each school.

Services	Description	Local	Central
Apache	Webserver allows all users to create a website		
CUPS	Print server. The aim is that it will also manage print quotas		
DHCP	Automatic connection of computers in the network		
DNS	Name server		
LDAP	Directory server containing user data for logon, file sharing and group information		
LTSP	Thin client server		
NFS	Network file system		
NTP	Clock server so that all machines have the correct time		
SMTP/IMAP over SSL	Email to everyone locally at the school		
SSH	Remote control over encrypted connection		
Squid	Cache for web sites (to save bandwidth)		
Webmin	System management via the web browser		
User administration	Simplified user administration		
Backup	Backup (should be done on a separate machine)		
SMB	Samba for connecting Windows-computers		
cfengine	Automatic control of the system setup		
Host and service monitoring	Monitoring the health of the server machine		
Appletalk	Connecting Macs		
SQL-server	Supplied (not set up)		

6.5 Use of resources in operations

For day-to-day operation of its computers, each school has an ICT contact. ICT contacts have from 2 to 4 hours allocated to this work per week. In addition, the municipality has an ICT tutor in a 50% position working with, among other things, competence and operations. The operation of the Linux network would gradually be transferred to the municipal ICT service in the school year 2005-2006.

On behalf of Debian Edu, Kapp næringshage (business park) made a calculation program which estimates the use of resources for ICT in schools. Today we operate the school network for over 3000 users with 2.1 FTEs. (Schools' ICT contacts allocate a total of 1.6 FTEs for their work, and 0.5 FTEs in the municipality.) When Kapp næringshage calculates our resource requirements for operation, they estimate the current needs to be 4.6 FTEs. This shows that the school has managed much with few resources.

Increased resources must primarily be placed in schools, because the schools' ICT contacts get busier when the number of PCs goes up. Increased number of pcs means increased use, and increased need for guidance in educational use of ICT tools.

Maintenance will increase faster than number of simultaneous users, but maintenance of the machines themselves will increase almost linearly with the number of machines. We want to focus more on the educational use of the equipment, and want most of the increase to be go towards use of ICT tools in school subjects.

There will also be a somehow greater need for increased resources for the municipality's ICT service, but because of economies of scale, the increase here will be small.

Today it is difficult for schools to prioritize hours to an ICT contact. Both because money for this must be taken from already stressed school finances, and because schools have lacked guidance on what and how much ICT contacts at schools will perform.

6.5.1 Operations' roles

The duties of ICT contact at each school:

- Oversee the school's server room.
- Be the school's contact at the municipality - report errors and outages.
- Perform simple maintenance tasks such as replacing mice and keyboards, upgrading thin clients, and simple patching.
- Be the school's superuser - able to advise colleagues about: user interfaces, e-mail, video projectors and relevant applications.
- Participate in ICT gatherings.
- Create and administer local users.
- Perform simple maintenance of printers.
- Create and manage email accounts.
- Facilitate the use of ICT in teaching.
- Perform simple commands and operations under guidance of an ICT-tutor.

From experience, we reckon these tasks take a minimum of 4 hours a week for a school with 50 thin or diskless clients. If the school has fewer machines, this reduces the number of hours a bit. With more machines, for example 150 machines, the local ICT contact at each school needs around a 30% position to easily handle technical maintenance.

If the school can't set aside a sufficient number of hours to the ICT contact, duties in the work list above must be removed, and the opposite if the school can use more hours.

Tasks beyond this, for example updating a website, being an instructor (beyond normal collegial guidance), must be agreed individually for compensation/taking time off.

The ICT supervisor recommends the following tasks for ICT and ICT service supervisor.

Operations:

- Mentor ICT contacts by telephone and e-mail.
- Visit the school for troubleshooting defects and errors on computers, printers and servers.

- Make joint purchases of computer equipment and enter into joint agreements etc.
- Backups.
- Continuous updating of software on school servers.
- Procurement of equipment and software with tenders in the market.

Skills:

- Develop the competence plan.
- Provide schools with courses in the educational use of ICT.
- Operations course.
- Training of ICT contacts in schools.
- Introduction to user interface and standard programs for teachers.

How much of the central operating resources are required, depends on which client types you have selected. Operation of workstations is almost twice as expensive for the operation of diskless clients.

6.5.2 Operation and support costs

The definition of operating costs:

- All activities and initiatives to deliver and/or maintain the desired use of ICT infrastructure.

We have described what is a realistic operating environment, considering a moderate level of service with proactive operation. The Norwegian "Program for digital kompetanse" is the basis for our assessments.

Proactive operation is to discover and rectify errors before they affect users. An example of proactive operation is to update laptops with new disk images once a week. When teachers log in the morning after, all the machines will have been reset to the school's preferences.

Operations get messages about defects in the system before it goes wrong for users. Defects are rectified and bugs fixed before users notice anything. A system example that provides messages used to proactive operation is disk store. They can notify when a hard disk is defective, or if the disk cache is full. Operations can also get information if the computer network is available, or whether processes must be terminated when users logs out.

- Advantage: One achieves very high stability of the system, supposing one has access to the right tools and the right skills. It becomes easier to maintain multiple types of computers because they know if they work or fail and can replace faulty equipment. Disadvantage 1: Requires higher technical expertise. Higher costs of establishing and managing operations. Disadvantage 2: Proactive operation is more costly than reactive operation if one does not count the loss of working hours for equipment that is defective. What you focus on depends on what the consequences are if the system is down. It is difficult to calculate the loss of teaching when ICT tools do not work. If it is required that the pupils and teachers have little downtime, one must invest in high uptime.

When we expand the fleet in schools this must have an impact on both ICT contacts' working resource and municipal ICT service for schools.

To quantify the need, we have estimated increased resource requirements in some of our investment options:

Investments	Servers	The number of clients/portables	Users:	Stipulated resource needs in 2008	Today's real resource:
Today's need 2005:	11	506	More than 3000	4.6 FTEs (man-year)	2,1 FTEs
Pupils in 2008	29	1033	More than 3000	6,9 FTEs	

6.5.2.1 Teachers in 2008

Alternative 1	280	266	4,3 FTEs	
Alternative 2 (laptop)	266	266	5,9 FTEs*	

-) Additional FTEs for maintenance of 266 laptops

Costs for the administration of all the computers for students and teachers. We go out from Alternative 1 in regards to the thin clients for students and teachers, and some laptops.

Year	***Number of PCs***	***Central operator***	***ICT guide for the entire municipality***	***ICT-contact at each school (average)***	***In total***
2005	506	1/2 FTE	1/2 FTE	8,5 % FTE(3:30 hours per week)	2,1 FTEs
2005	**Human resources' costs for operations***			**NOK 980 910,-**	
2008	1333	1 position	1/2 FTE	100 % FTE(26 hours per week)	11,5 FTE
2008	**Human resources' costs for operations***			**NOK 5 400 00,-**	

-) NOK 270, - per teacher hour 1730 hours a year. ICT contact at each school uses 75% of the time on pedagogical support.

Alternative 2 with laptops for every teacher:

Year	***Number of PCs***	***Central operator***	***ICT guide for the entire municipality***	***ICT-contact at each school (average)***	***In total***
2008	1333	1 + 4/5 FTE*	1/2 FTE	100 % FTE(26 hours per week)	12,8 FTEs
2008	**Human resources' costs for operations***			**NOK 6,000,000.-**	

-) An additional position for maintenance of laptops.

Training costs for students and teachers is roughly the same with Windows and Linux, according to surveys done in schools in Norway and the UK. This is because the training is related to the use by end user programs in everyday school life.

- Usually, at a school with 300 students and teachers only one or two persons need training in operating computer systems. This refers to both an ICT contact at school, and an operator in the municipality.

We have included additional training costs for Linux. When all teachers have one day going through the Linux desktop option, the transition to the new system goes easier for those who think they can only manage Windows. The cost for displays like LærerIKT and similar are not included as we have done in the cost overview from the municipalities in this survey.

6.6 Summary of the options

To achieve the objective of one computer per third student and one pc per teacher Option 1 is recommended. This option requires over 800 additional computers compared to current coverage of 506 client machines. Overall, we will then get just over 1300 client computers with emphasis on thin and diskless clients. Some machines will also be portable to provide additional flexibility in everyday school life.

Alternatives	**Cost over 3 years**
Option 1: Development of 800 clients	NOK 7,322,000.-
Option 2: Portable to all teachers + Option 1.	NOK 12,000,000.-

Operating costs:

Alternatives	**Yearly cost**
Alternative 0. Administrating 506 client machine	NOK 1,000,000.-
Option 1. Increases with about 800 to 1300 PCs	NOK 5,400,000.-
Option 2. Portables to to all teachers*	NOK 6,000,000.-
Option 3. Portable room layout*	NOK 8,000,000.-

The large increase in operating expenses from current option 0 to option 1 is due to investment in ICT contact at schools. This increases from 10% to a full 100% FTE. The maintenance ICT contacts do today is around 10% FTE. This will probably increase to 20% with a doubling of the number of client machines. On increase to a full-time position, 80% will be used to support the educational use of ICT tools in school subjects. This means that principals at school must allocate resources for this, so that one follows the national curriculum from 2006.

6.7 Recommendation

Alternative 1:

Type of cost	**Amount**
Of this is the administration of 1300 client computers:	NOK 2,000,000.-
Annual investment in three years:	NOK 2 440 667,-
Support for educational use of ICT tools:	NOK 3,400,000.-
Annual cost for Option 1 with investment and operations:	NOK 7,841,000.-

6.8 Attachment

Many schools have developed an activity plan for the use of ICT in the school. This should be included as attachment.

7 Setting up infrastructure

7.1 Network architecture

User case: Shall set up a computer network that scales so that one can either operate the system locally or connect to a centralized operational solution

7.1.1 Solution

7.1.2 Exception handling

7.1.3 Verification

7.1.4 Update the configuration database

7.2 Server profiles

Use case: How to install machines for an entire computer network for a school or many schools in a municipality.

  {{attachment:bilder50.png}} **The different profiles on different servers.**

7.2.1 Combi-server as a combined resolution

Two profiles with main server and thin client server in combination are called a combi-server

- {{attachment:bilder51.png}}

This is a fairly small step, which makes it easy to use an appropriate switch on the backbone network, and use a crossover cable to connect the firewall with a combi-server

Note: Be aware that setting a printer on the address 192.168.0.0/24, which is the thin client network, does not work if the hostname is **printer00**. Be sure to edit KDE printing manager to search for printers at 192.168.0.0/24-net. Not standard 10.0.2.0/23-net.

7.2.2 Description of the profiles in Skolelinux/Debian-Edu

Profiles shown during the installation originate from the file `src/debian-edu-install/debian/debian-edu-install.template` in the debian-edu-install package.

Graphical desktop

One will increasingly see references to a graphical desktop. In short that means a modern desktop with point and click, windows, icons, and file folders. Graphical user interfaces were first made by Xerox Parc in 1973, 10 years before they came to personal computers that could be bought on the market. This was a *very* short presentation of graphical user interfaces.

A brief summary of the different profiles in Skolelinux/Debian-edu and how they can be combined

<ol style="list-style-type: decimal;"> Main server Warning: *All* Skolelinux/Debian-edu-networks *must have* only one main server, and only one machine with that profile. Most commonly that profile can be combined with thin client servers, or just a workstation. Every Skolelinux network needs one, and only one machine running the 'Main Server'. This machine provides network services such as for example network login with the help of directory server (LDAP) etc. Without this machine the network does not work. Since this machine will save all data files, it needs a lot of disk space. You do not get graphical user interface by installing this profile. If you want a graphical user interface you must also install workstation-profile or thin client server.   Workstation

Machines running the 'workstation' profile is what we know as normal PCs. Users log on to workstations, and get storage space on The main server. Documents, personal settings and many network services are on The main server. User programs run on the workstation.

For access to CD/DVD-player/burner, digital cameras, scanners, this is the profile to install.

1. Thin client server

Machines running the thin client server support the thin clients. This profile also includes the workstation profile. To prevent saturating the network, two NICs are required. The profiles: main server, workstation and thin client server can be installed on the same machine.

This profile also contains the work station-profile

1. Diskless workstations

Machines running as thin client servers provide support for diskless clients if this is enabled. In Skolelinux 2.0 this must be enabled afterwards. This profile also includes the workstation-profile. Profiles main server, workstation and thin client server can be installed on the same machine.

- This profile also contains the work station-profile
- Main server + thin client server (including workstation)

This combination of profiles, also called combined profile, providing the ability to setup a complete Skolelinux / Debian-edu network workstations and thinclients with only one server. This is an acceptable solution in a small Skolelinux/Debian-edu network, with maybe 10-15 thin clients and a few workstations. For larger installations, one must usually choose servers which are *larger*.

1. Main server + workstation

This combination of profiles mainly gives you a main server with a GUI. If you do not like the idea of administering your main server from the command line, this is a good combination.

The standalone profile is not part of Skolelinux/Debian-edu network. The purpose of this profile is to support the home PC or portables.

1. Stand alone

The standalone profile can not be installed together with the main server, workstation or thin client server.

The standalone profile is best to use without linking it to a Skolelinux / Debian-edu network.

All programs in Skolelinux/Debian edu are included in the standalone profile

7.2.3 Solution

7.2.4 Exception handling

7.2.5 Verification

7.2.6 Update the configuration database

7.3 Hardware servers

User case: What's should be configured

7.3.1 Solution

7.3.2 Exception handling

7.3.3 Verification

7.3.4 Update the configuration database

7.4 Client computers

User case: Choice of client machines. Should you choose silent machines or machines for multimedia. Should one have laptops to all or desktops.

Several types of technologies can provide application on the PC. Most common is thick clients operating locally on each computer. But there are other types of technology for applications on the desktop. Many have heard of graphic terminals. Examples include Citrix, !FreeNX and Windows Terminal Server. There are also other options like lowfat clients and real thin clients. This article describes the options and provides an overview of where the various terminal technologies do best. The reason for the article is the experience of enterprise solutions with centralized operation of the computer in many different buildings with low, medium or high network capacity.

Client technologies are described in the following order. Graphic terminals Citrix and !FreeNX, thin clients with X Windows, thick clients with Linux and Windows, client in between with Linux, and laptops. The following are examples of what server systems are commonly used in various business-oriented installations. A key factor for calculating costs is the number of concurrent users and the number of servers. Centralized management of computer equipment at several schools may in practice be compared with how the operations of ICT systems is done in larger companies. Often schools have more computers than the rest of the council's activities. Failure to think things through in what one chooses for client solutions in schools can quickly lead to a doubling of the number of employees in IT services in the municipality.

Citrix is the most known product for **graphical clients**. The company making this is product was established in 1989. The first graphical clients were made for the operating system OS/2. First Windows product was launched with NT 3.51 in 1995. There are several competing products to Citrix. One of the most successful is the NX technology. Briefly, you may run applications from a server with Citrix or NX. The screen is exported over the network from a server to a graphical terminal on a thick client.

Graphical clients have the strength that it would be seemless what ever kind of operating system that might be running on the client. One could use the applications on the server anyways. One can run standard office programs and client emails over an ISDN line with 64 kbps. That said, there are limitations in graphic software, whether it is used with multimedia or interactive graphics. The solution can quickly become of no practical use if a municipality distributes 30 or 50 graphic terminals at 5-6 schools with broadband with 2-8 Mbps. With this capacity one can not run interactive graphical applications. The Internet would be filled up with traffic, and the Citrix client would disconnect from the server machine.

With **graphical clients** the operations department must run two parallel paths for the maintenance of software. Maintenance occurs on all client computers and on local and central servers. For getting for example Citrix to work reasonably well, there must be deployed two additional server machines in each building, in addition to central application servers. In addition, it usually needs some thick clients also for use with multimedia. For example 1/3 of the machines in Oslo schools are thick clients to provide support for multimedia.

Thin clients was introduced in 1984 at MIT. This was around the same time Apple released the Macintosh GUI. The following year Microsoft shipped the first edition of MS-Windows. Actually thin clients are named X Window Systems and can be used on all possible platforms like Linux, Mac or Windows. X Windows turned things upside down. In practice applications run on a server, and the GUI is sent over the network to the client computer. The client computer runs a server program to display graphical windows. An X server may run your application windows from different programs running on many different servers. Thick clients also run the X Window system, using a virtual local network on the PC. All Unix systems with graphical user interfaces run X servers.

The main advantage of **thin clients** is the reuse of older equipment without increasing the complexity of operations. Many people use PCs with 233 MHz and 32 MB memory as thin clients. There is no need for local hard drive. Users can handle heavier graphics, sound and simple video. Several schools have opened up for the use of CD / DVD-Rom and USB memory stick at **the thin clients**. Operating personnel do not have to keep track of a separate operating system on each of the PCs. Everything is handled from the server. Each thin client uses around 2 Mbps network capacity during normal use. The performance of thin clients is significantly better than graphic terminals. Thin clients need in average fewer servers than graphic clients with for example Citrix, as shown by a study of The Department of Education in Oslo.

Thick clients or standard PCs is what is mostly used today. The term Personal Computers were used for the first time November the third 1962. The first PC with network and graphical user interface was created at Xerox PARK in 1973. Today it is the PC concept IBM launched in 1981 that is known and widespread. The entire operating system and all the software applications are installed on each client computer on a local data store. The most famous operating computers are Microsoft Windows and Linux. But there are also a number of other systems that many people use, including a version of BSD.

The advantage of **Thick clients** is that all programs are run locally, which can provide great flexibility and performance for users. Since most user programs run locally few central servers are needed. Solutions with thick clients can be relatively inexpensive to operate if one standardizes. On Windows, it is a great advantage to have mostly similar machines, which is difficult over time. It is quite common, for example, that the school has both 4 and 5 PC types. This affects operational

costs. Linux is more flexible because the system can be more easily managed with many different PC types. Linux also requires less memory, and allows for longer use of older computers without loss of performance as the British Educational Communications and Technology Agency (BECTA) reports.

Diskless clients is another exciting technology. Today supported on Linux with Lessdisks or new LTSP. Novell had a virtual monopoly on diskless clients 15 years ago. Simply explained, the entire operating system and applications are installed on a server. The operating system is uploaded from the server to the client over the network. File, print, and Web services are handled by an operating system designed for networks. By the introduction of Windows 95, Novell met a technological barrier. Microsoft changed controlling Windows with registry instead of text files. Now it's only Linux and other Unix variants offering diskless clients.

The advantage with **diskless clients** is that you get the performance of thick clients with the operational advantage of thin clients. It means that the organisation can connect many client machines to a server, without installing locally an operating system on each client. Everything is handled from the server. The system supports audio, video, CD/DVD-Rom and USB memory stick. Today it is unusual to find used machines with less than 800 MHz processors and 256 MB of memories, which is well suited for the half thick clients. It is recommended to use local hard drive cache.

Portable machines are essentially thick clients. Laptops may in principle be used as thin clients, half thick clients or graphical terminals. But it is not very practical for several reasons. Portables should be used as thick client. In order to connect the laptop to a stationary computer network, one must choose what kind of services to be used.

There are significant challenges with portables in wireless networks with many users. Wireless networks have limited capacity. Portables are also subject to rough treatment, and require more frequent replacement than what is normal for stationary equipment. One should not run graphical terminals on laptops in wireless networks. This quickly becomes unstable when you have many users. Thick clients with Linux or Windows run fine. They can relatively easily be authenticated against the network. The user can access file directories, printing and other network services in a safe and secure manner. Several providers offer laptops in schools which connect to the computer network running Debian Edu.

7.4.1 Table of client types

Main solution	***Support for multimedia***	***Characteristics***
Fat clients (Windows, Linux or Mac)	Good support for sound, graphics and video with powerful enough processor and memory on the client machine.	All user applications installed on the client machine. The user programs run on the client machine. The client machine may be stationary or portable. Running multiple services in networks such as email, file storage, case-filing system etc.**Advantage:** Requires few server machines. Good support for multimedia**Disadvantage:** Need to install and maintain all the software on each client machine
Diskless workstation (Linux. Earlier this was the solution from Novell with Windows 3.X)	Good support for sound, graphics and video given a powerful enough processor and memory on the client machine.	All user applications are installed on the server machine. User programs run on the client machine. Client computer is usually stationary. Running multiple services on the network such as email, file storage, case-filing system etc.**Advantages:** Same functionality as thick clients. Need few servers. The client computers do not have software installed.

Thin client (X Window System)	Decent audio, graphics and video support given a powerful enough processor memory on the server machine. Needs high capacity client network.	All user programs and services are installed on the server machine. The user programs running on servers. The client computer is usually stationary. Running multiple services in networks such as email, file storage, case-filing system etc.**Advantage:** Gives new life to reused computers. Client does not have installed software.**Disadvantage:** Requires more servers than thick and diskless clients.
Graphical terminals (FreeNX, Citrix, RDP)	Decent graphics support given powerful enough processor memory on the server machine, and high capacity network. Weak or little support for interactive graphics at medium capacity network.	All user programs and services are installed on the server machine. A full operating system with a graphical interface is usually installed on the client machine. The user programs run on the server. The client computer is usually stationary. Several network services such as email, file storage, case-filing system etc. are provided.**Advantage:** Gives new life to reused computers.**Disadvantage:** Must install and maintain the operating system on each client machine. Requires more servers than real thin clients. Requires significantly more servers than thick or diskless clients. Gives poor performance or no support for multimedia. The terminal disconnects with network overloads. This may happen several times an hour.
Laptops	Good support for sound, graphics and video with powerful enough processor and memory on the client machine.	**Advantage:** Can take the PC anywhere suitable**Disadvantage:** Must install and maintain the operating system on each client machine. Must set up and maintain services that make it easy to connect and disconnect machines on the network. There is considerable breakage with portable equipment, and lifetimes average 3 years; that's 2-5 years less than desktops. Administration of portable devices is expensive.

7.4.2 Solution

7.4.3 Exception handling

7.4.4 Verification

7.4.5 Update the configuration database

7.5 Switches

User case: What's should be configured

7.5.1 Solution

7.5.2 Exception handling

7.5.3 Verification

7.5.4 Update the configuration database

7.6 Wireless access points

User case: What's should be configured

7.6.1 Solution

7.6.2 Exception handling

7.6.3 Verification

7.6.4 Update the configuration database

7.7 Firewall(s)

User case: What's should be configured

7.7.1 Solution

7.7.2 Exception handling

7.7.3 Verification

7.7.4 Update the configuration database

7.8 Routers

User case: What's should be configured

7.8.1 Solution

7.8.2 Exception handling

7.8.3 Verification

7.8.4 Update the configuration database

7.9 Setting up a simple firewall

User case: What's should be configured

7.9.1 Solution

7.9.2 Exception handling

7.9.3 Verification

7.9.4 Update the configuration database

7.10 Setup:

User case: What's should be configured

7.10.1 Solution

7.10.2 Exception handling

7.10.3 Verification

7.10.4 Update the configuration database

8 Useful commands

8.1 Support for 4 GB memory <-- included in configuration management

Use Case: Because there is limited space on the Skolelinux/Debian-Edu CD only one Linux kernel is included, i.e. the lowest common denominator. That means a kernel *working* at as many as possible different types of hardware is included.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

You can find out the type of kernel running with the command **uname -a**. The command can be used later to ensure that you have upgraded to the required core. Then it may look like this:

```
tjener:~# uname  -a
Linux tjener.intern 2.6.8-2-386 #1 Thu May 19 17:40:50 JST 2005 i686 GNU/Linux
```

Here runs a 386-core, which should work on just about all of the PCs. But it is not optimal for dual core processors or more than 940MB.

If you want a kernel for new servers with plenty of memory and multiple processors, you can download and install it afterwards. Debian's package system makes that easy.

Look at Section 8.9 for a more detailed description of **apt-get** and **dpkg**.

smp is the keyword to look for when you want a Linux kernel with support for more RAM than 940MB of memory and dual processors. The acronym stands for *Symmetric Multi-Processors*. The command is run from a shell which lists the number of cores ready for installation:

apt-cache search kernel-image | grep smp

. At the time this is written the following is listed:

```
kernel-image-2.4-686-smp - Linux kernel image for version 2.4 on PPro/Celeron/PII/PIII/P4 ↩
    SMP
kernel-image-2.4-k7-smp - Linux kernel image for version 2.4 on AMD K7 SMP
kernel-image-2.4.27-2-686-smp - Linux kernel image for version 2.4.27 on PPro/Celeron/PII/ ↩
    PIII/P4 SMP
kernel-image-2.4.27-2-k7-smp - Linux kernel image for version 2.4.27 on AMD K7 SMP
kernel-image-2.6-686-smp - Linux kernel image for version 2.6 on PPro/Celeron/PII/PIII/P4 ↩
    SMP.
kernel-image-2.6-amd64-k8-smp - Linux kernel image for version 2.6 on AMD64 SMP systems
kernel-image-2.6-em64t-p4-smp - Linux kernel image for version 2.6 on Intel EM64T SMP ↩
    systems
kernel-image-2.6-k7-smp - Linux kernel image for version 2.6 on AMD K7 SMP.
kernel-image-2.6.8-11-amd64-k8-smp - Linux kernel image for version 2.6.8 on AMD64 SMP ↩
    systems
kernel-image-2.6.8-11-em64t-p4-smp - Linux kernel image for version 2.6.8 on Intel EM64T ↩
    SMP systems
kernel-image-2.6.8-2-686-smp - Linux kernel image for version 2.6.8 on PPro/Celeron/PII/ ↩
    PIII/P4 SMP.
kernel-image-2.6.8-2-k7-smp - Linux kernel image for version 2.6.8 on AMD K7 SMP.
```

There is no need to specify a specific kernel version like 2.4.27 or 2.6.8. Just use 2.4 or 2.6. This boils down to

```
kernel-image-2.4-686-smp - Linux kernel image for version 2.4 on PPro/Celeron/PII/PIII/P4 ↩
    SMP
kernel-image-2.4-k7-smp - Linux kernel image for version 2.4 on AMD K7 SMP
kernel-image-2.6-686-smp - Linux kernel image for version 2.6 on PPro/Celeron/PII/PIII/P4 ↩
    SMP.
kernel-image-2.6-amd64-k8-smp - Linux kernel image for version 2.6 on AMD64 SMP systems
kernel-image-2.6-em64t-p4-smp - Linux kernel image for version 2.6 on Intel EM64T SMP ↩
    systems
kernel-image-2.6-k7-smp - Linux kernel image for version 2.6 on AMD K7 SMP.
```

Now you just need to know what kind of processor you have eg. 686 (Intel), k7 (AMD) AMD64 or EM64T

As soon as which kernel fit the machine best it can be installed using the command

apt-get install kernel-image-2.6-<your processor type>-smp

If the machine contain an Intel Xeon one can use

apt-get install kernel-image-2.6-686-smp

If a 2.4 kernel is used

apt-get install kernel-image-2.4-<your processor type>-smp

With an AMD Athlon(TM) MP 2000 it is possible to use

apt-get install kernel-image-2.6-k7-smp

When you install the new kernel, you may see something like this:

```
tjener:~# apt-get update
tjener:~# apt-get install kernel-image-2.6-686-smp
Reading Package Lists... Done
Building Dependency Tree... Done
The following extra packages will be installed:
  kernel-image-2.6.8-2-686-smp
```

```
Suggested packages:
  lilo kernel-doc-2.6.8 kernel-source-2.6.8
Recommended packages:
  irqbalance
The following NEW packages will be installed:
  kernel-image-2.6-686-smp kernel-image-2.6.8-2-686-smp
0 upgraded, 2 newly installed, 0 to remove and 0 not upgraded.
Need to get 15.3MB of archives.
After unpacking 44.9MB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://ftp.debian.org sarge/main kernel-image-2.6.8-2-686-smp 2.6.8-16 [15.3MB]
Get:2 http://ftp.debian.org sarge/main kernel-image-2.6-686-smp 101 [2154B]
Fetched 15.3MB in 1m13s (208kB/s)
Selecting previously deselected package kernel-image-2.6.8-2-686-smp.
(Reading database ... 80762 files and directories currently installed.)
Unpacking kernel-image-2.6.8-2-686-smp (from .../kernel-image-2.6.8-2-686-smp_2.6.8-16_i386 ↩
    .deb) ...
Selecting previously deselected package kernel-image-2.6-686-smp.
Unpacking kernel-image-2.6-686-smp (from .../kernel-image-2.6-686-smp_101_i386.deb) ...
Setting up kernel-image-2.6.8-2-686-smp (2.6.8-16) ...
File descriptor 3 left open
File descriptor 4 left open
File descriptor 5 left open
File descriptor 6 left open
File descriptor 7 left open
    Finding all volume groups
    Finding volume group "vg_data"
    Finding volume group "vg_system"
Searching for GRUB installation directory ... found: /boot/grub .
Testing for an existing GRUB menu.list file... found: /boot/grub/menu.lst .
Searching for splash image... none found, skipping...
Found kernel: /boot/vmlinuz-2.6.8-2-686-smp
Found kernel: /boot/vmlinuz-2.6.8-2-386
Updating /boot/grub/menu.lst ... done
Setting up kernel-image-2.6-686-smp (101) ...
```

As shown, you were asked to install kernel-image-2.6-686-smp, and it automatically translated to install kernel-image-2.6.8-2-686-smp. It was also suggested to install some other packages that might be useful.

Reboot the machine with the command: shutdown -r now

8.1.1 Exception handling

To activate a new kernel the machine need to be rebooted.

Building the kernel for a Skolelinux / Debian-Edu machine, is the only time you ever need a restart. When installing other programs there is no need for a restart.

8.1.2 Verification

When running the command **uname -a** after installation, the following is displayed

```
tjener:~# uname  -a
Linux tjener.intern 2.6.8-2-686-smp #1 SMP Thu May 19 17:27:55 JST 2005 i686 GNU/Linux
```

After the installation of the smp kernel and after the reboot, you can run the command **free** and **cat /proc/cpuinfo**. Then you can see if the new kernel uses all memory and both processors.

```
ltspserver00:~# free
             total       used       free     shared    buffers     cached
```

```
Mem:       4074752    4045556      29196          0     339248    2327780
-/+ buffers/cache:    1378528    2696224
Swap:      1835000       5852    1829148
```

Here is a shortened printing with the unnecessary printing removed .

```
ltspserver00:~# cat /proc/cpuinfo
processor       : 0
vendor_id       : !GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Xeon(TM) CPU 2.66GHz

processor       : 1
vendor_id       : !GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Xeon(TM) CPU 2.66GHz

processor       : 2
vendor_id       : !GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Xeon(TM) CPU 2.66GHz

processor       : 3
vendor_id       : !GenuineIntel
cpu family      : 15
model           : 2
model name      : Intel(R) Xeon(TM) CPU 2.66GHz
```

8.1.3 Update the configuration database

8.2 Administrating packages (apt-get)

Use case: Installing new programs or update programs

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

To install packages one needs to tell from where they should be fetched. In other words, which package archive to use.

One can specify package archives in the file `/etc/apt/sources.list`

You can work with package management on the command line. There is more graphical applications like for example KPackage 7 or Webmin 12

This section provides a quick introduction to using the command line for administrating packages.

This is the content of a file with references to package repositories on the Internet or from a CD ROM:

```
#deb file:///cdrom/ sarge main local

deb cdrom:[Debian GNU/Linux edu _Sarge_ - Unofficial i386 Binary-1 (20050808)]/ unstable ↩
    contrib local main non-free

 1. deb http://security.debian.org/ stable/updates main contrib non-free
 1.deb http://security.debian.org/ sarge/updates main contrib non-free
   1. Use (by uncommenting) either http or ftp, NOT both
   1. http based apt source: ----------------
 1. deb http://ftp.debian.org/debian/ sarge main contrib non-free
```

```
 1. deb http://non-us.debian.org/debian-non-US/ sarge/non-US main contrib non-free
 1. deb http://ftp.skolelinux.no/skolelinux/ sarge local
   1. ftp based apt source: -----------------
 1. deb ftp://ftp.debian.org/debian/ sarge main contrib non-free
 1. deb ftp://non-us.debian.org/debian-non-US/ sarge/non-US main contrib non-free
 1. deb ftp://ftp.skolelinux.no/skolelinux/ sarge local
```

Note that the lines *without* hashtag (#) can be used as reference to the package archive. The example shows that one only gets packets from the CD ROM used during installation. Other archives are not activated. When doing this, one should open for security upgrades. So you can try other archives for more packages.

As a start it should look something like this:

```
#deb file:///cdrom/ sarge main local

 1.deb cdrom:[Debian GNU/Linux edu _Sarge_ - Unofficial i386 Binary-1 (20050808)]/ unstable ↩
      contrib local main non-free

 1.deb http://security.debian.org/ stable/updates main contrib non-free
deb http://security.debian.org/ sarge/updates main contrib non-free
   1. Use (by uncommenting) either http or ftp, NOT both
   1. http based apt source: ----------------
deb http://ftp.debian.org/debian/ sarge main contrib non-free
deb http://non-us.debian.org/debian-non-US/ sarge/non-US main contrib non-free
deb http://ftp.skolelinux.no/skolelinux/ sarge local
   1. ftp based apt source: -----------------
 1. deb ftp://ftp.debian.org/debian/ sarge main contrib non-free
 1. deb ftp://non-us.debian.org/debian-non-US/ sarge/non-US main contrib non-free
 1. deb ftp://ftp.skolelinux.no/skolelinux/ sarge local
```

Note that there is a # sign in front of the line containing "deb: cdrom". There is no need to load packages from a CD-ROM when one can get everything from the Internet.

If you add a new line to this file, you must update the database with information about what that is available.

See chapter 13 for other lines to add as package sources.

8.2.1 Exception handling

Links to package archives have a specific form. Failure to follow this gives error messages when updating, asking to correct the error.

The comment sign (#) is also in place in front of several lines in the file. The technique of "commenting out" is typical for most configuration files in Linux. Other symbols to be used is the semicolon (;) and double slashes (//). But here, the hashtag is in force, and when removed, what is written on the line is operative.

8.3 Update the package archive

Use case: Update the package repository with a summary of updated programs.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

The selection of available packages is constantly updated. The most common is new security updates. New versions of the software can also be posted. Therefore, one must update the package archives. This is done with the following command

```
tjener:~# apt-get update
Get:1 http://ftp.skolelinux.no sarge/local Packages [17.4kB]
Ign http://ftp.skolelinux.no sarge/local Release
Get:2 http://non-us.debian.org sarge/non-US/main Packages [20B]
```

```
Get:3 http://non-us.debian.org sarge/non-US/main Release [102B]
Get:4 http://non-us.debian.org sarge/non-US/contrib Packages [20B]
Get:5 http://non-us.debian.org sarge/non-US/contrib Release [105B]
Get:6 http://non-us.debian.org sarge/non-US/non-free Packages [20B]
Get:7 http://non-us.debian.org sarge/non-US/non-free Release [106B]
Get:8 http://ftp.debian.org sarge/main Packages [3347kB]
Get:9 http://security.debian.org sarge/updates/main Packages [155kB]
Get:10 http://security.debian.org sarge/updates/main Release [110B]
Get:11 http://security.debian.org sarge/updates/contrib Packages [538B]
Get:12 http://security.debian.org sarge/updates/contrib Release [113B]
Get:13 http://security.debian.org sarge/updates/non-free Packages [20B]
Get:14 http://security.debian.org sarge/updates/non-free Release [114B]
Get:15 http://ftp.debian.org sarge/main Release [95B]
Get:16 http://ftp.debian.org sarge/contrib Packages [56.2kB]
Get:17 http://ftp.debian.org sarge/contrib Release [98B]
Get:18 http://ftp.debian.org sarge/non-free Packages [58.4kB]
Get:19 http://ftp.debian.org sarge/non-free Release [99B]
Fetched 3635kB in 23s (157kB/s)
Reading Package Lists... Done
```

This command must be executed *before* an upgrade or before adding new packages.

8.3.1 Exception handling

8.3.2 Verification

8.4 Update to new packages

Use case: Updating the installed packages to a newer version if one is available

Author: Klaus Ade Johnstad

Co-author: Knut Yrvin

All installed packages can be upgraded to newer versions using the command

apt-get upgrade

```
tjener:~# apt-get upgrade
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be upgraded:
  apache apache-common apache2-utils bsdutils cfengine cfengine-doc courier-authdaemon ↩
      courier-base courier-imap courier-imap-ssl courier-ldap
  courier-ssl cpio debian-edu-config debian-edu-install education-common education-main- ↩
      server education-networked education-tasks libapr0 libice6
  libmysqlclient12 libpam-ldap libpcre3 libsensors3 libsm6 libsnmp-base libsnmp5 libssl0 ↩
      .9.7 libungif4g libx11-6 libxext6 libxft1 libxi6 libxmu6 libxmuu1
  libxp6 libxpm4 libxrandr2 libxt6 libxtrap6 libxtst6 localization-config lynx mount mysql- ↩
      common ntp ntp-refclock ntp-server ntpdate openssl python2.3
  slbackup snmp squid squid-common tcpdump util-linux xdebconfigurator xfree86-common xlibs ↩
       xlibs-data
62 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 23.7MB of archives.
After unpacking 225kB disk space will be freed.
Do you want to continue? [Y/n]
```

Just press **Enter** or 'Y' and **Enter**. The packages will be downloaded and installed automatically. One will get a change log when the upgrade begins.

Once you have upgraded, you can delete the packages downloaded in the directory`/var/cache/apt/archives/`. Use the command

apt-get clean

to clear the archive. This should be done occasionally. Otherwise `/var` becomes full.

8.4.1 Warning

Sometimes it's OK to see what is going to happen *before* upgrading. If judging whether it is necessary to download several large packages, maybe you need to wait until there is more bandwidth available. If you run

apt-get upgrade --simulate

if one will simulate what will happen, without it actually happens. Is it too much information on the screen, one can run

apt-get upgrade --simulate | more

If it looks good, one can run the command again without the --**simulate** parameter

It is also possible to use **aptitude dist-upgrade** in combination with **apt-get upgrade**.

8.4.2 Exception handling

Sometimes you will get a message about changes affecting packages to upgrade or install, as in here

```
kdeaddons (4:3.1.0-4) unstable; urgency=low

  * Rebuilt against libvorbis0a (closes: #184713).
  * Removed alpha compile flags.
  * Fresh admin/ sync.

 -- Ben Burton <bab@debian.org>  Sun, 16 Mar 2003 16:00:19 +1100

kdeaddons (4:3.1.0-2) unstable; urgency=low

  * First KDE3 upload to debian!
  * Applied Ewald Snel's patch for xine support.
  * Rolled the epoch to aid upgrades from the unofficial repository on
    ftp.kde.org.. *sigh*
```

Use **Space** on the keyboard to browse through the message. Then you will see

```
quanta (1:3.0pr1-1) unstable; urgency=low

  * New upstream release.
  * Built for KDE3.

 -- Ben Burton <benb@acm.org>  Wed,  4 Sep 2002 10:36:12 +1000

(END)
```

Press the **q** key to quit and you get

```
Fetched 60.2MB in 11m24s (87.9kB/s)
Reading changelogs... Done
apt-listchanges: Do you want to continue? [Y/n]?
```

To continue you need to press **Y** for Yes.

8.4.3 Verification

8.5 Summary of installed packages

Use case: Want a summary of installed packages

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

To get a summary of the installed packages run this command

dpkg --list | more

Note that when the first letters in the list is "ii" it mean the package is fully installed.

To get the status of one particular package one can use **grep** to search for it:

```
tjener:~# dpkg --list | grep apache
ii  apache         1.3.33-6       versatile, high-performance HTTP server
ii  apache-common  1.3.33-6       support files for all Apache webservers
ii  apache2-utils  2.0.54-4       utility programs for webservers
```

8.6 Find the name of a particular package

Use case: Often it is hard to remember the name of a package.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

To find a particular package one can use a search term with this command:

apt-cache search <package name>

Try this if there is too much text on the screen

apt-cache search <package name>|more

The two symbols < and > must *not* be used. They are only used in this example.

```
tjener:~# apt-cache search apache
apache - versatile, high-performance HTTP server
apache-common - support files for all Apache webservers
apache-dbg - debug versions of the Apache webservers
apache-dev - development kit for the Apache webserver
apache-doc - documentation for the Apache webserver
apache-perl - versatile, high-performance HTTP server with Perl support
apache-ssl - versatile, high-performance HTTP server with SSL support
apache-utils - utility programs for webservers (transitional package)
```

As the screen dump show there are a lot more related to apache than the packages already installed.

8.7 Show available information about packages

User case: Want to get information about the package. There may be dependencies to other packages etc..

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

The command

apt-cache showpkg <package name>

and

<apt-cache policy <package name>

gives details about the package.

```
tjener:~# apt-cache showpkg kdissert
Package: kdissert
Versions:
0.3.8-1(/var/lib/apt/lists/ftp.debian.org_debian_dists_sarge_main_binary-i386_Packages)

Reverse Depends:
Dependencies:
0.3.8-1 - kdelibs4 (2 4:3.3.2-4.0.2) libc6 (2 2.3.2.ds1-4) libgcc1 (2 1:3.4.1-3) libqt3c102 ↩
    -mt (2 3:3.3.3) libstdc++5 (2 1:3.3.4-1)
Provides:
0.3.8-1 -
Reverse Provides:
tjener:~# apt-cache policy  kdissert
kdissert:
  Installed: (none)
  Candidate: 0.3.8-1
  Version Table:
     0.3.8-1 0
        500 http://ftp.debian.org sarge/main Packages
```

So one notices that the package kdissert is not installed, but available for installation in version 0.3.8-1 from `http://ftp.debian.org sarge/main`

8.8 Installation of packages

Use case: Want to install a program or a program package.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

When you have found the package to be installed, run the command

apt-get install <package name>

If you want to see what happened during the installation you should run a simulation first with the command

apt-get install <package name> --simulate

```
tjener:~# apt-get install  aterm --simulate
Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
  aterm
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Inst aterm (0.4.2-11 Debian:3.1r0/stable)
Conf aterm (0.4.2-11 Debian:3.1r0/stable)
tjener:~# apt-get install  aterm
Reading Package Lists... Done
Building Dependency Tree... Done
The following NEW packages will be installed:
  aterm
0 upgraded, 1 newly installed, 0 to remove and 0 not upgraded.
Need to get 91.6kB of archives.
After unpacking 287kB of additional disk space will be used.
Get:1 http://ftp.debian.org sarge/main aterm 0.4.2-11 [91.6kB]
Fetched 91.6kB in 1s (71.0kB/s)
Selecting previously deselected package aterm.
(Reading database ... 32924 files and directories currently installed.)
Unpacking aterm (from .../aterm_0.4.2-11_i386.deb) ...
Setting up aterm (0.4.2-11) ...
```

8.9 Removal of installed packages

Use case: Wants to remove certain packages that will not be used.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

To find a specific package to be removed, use the commands listed above.

When you have found the name of the package run the command

apt-get remove <package name>

If you want to see what happens when you remove the package, you may simulate this with the command

apt-get remove <package name> --simulate

```
tjener:~# apt-get remove  aterm --simulate
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be REMOVED:
  aterm
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
Remv aterm (0.4.2-11 Debian:3.1r0/stable)
tjener:~# apt-get remove  aterm
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be REMOVED:
  aterm
0 upgraded, 0 newly installed, 1 to remove and 0 not upgraded.
Need to get 0B of archives.
After unpacking 287kB disk space will be freed.
Do you want to continue? [Y/n]
(Reading database ... 32936 files and directories currently installed.)
Removing aterm ...
```

8.10 Install a specific package version

User case: Want a specific version of a package. It can for example be a previous release of a program.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

When installing a package with the command

apt-get install <package name>

then the newest package is installed. Sometimes an older version instead of the newest version is wanted.

apt-get install <package name>=older_version_number

To get an older version of the Webmin backup module one can run

apt-cache showpkg webmin-slbackup

to get a summary of the available version

```
tjener:~#  apt-cache policy webmin-slbackup
webmin-slbackup:
  Installed: 0.0.10-1
  Candidate: 0.0.10-1
  Version Table:
 *** 0.0.10-1 0
        500 http://ftp.skolelinux.no sarge/local Packages
        100 /var/lib/dpkg/status
     0.0.9-1 0
        500 http://ftp.debian.org sarge/main Packages
```

Here one can see that there are two versions available. Both 0.0.9-1 and 0.0.10-1

If the 0.0.9-1 version of the program is wanted it can be installed using the following command

apt-get install webmin-slbackup=0.0.9-1

```
tjener:~# apt-get install webmin-slbackup=0.0.9-1 --simulate
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be DOWNGRADED:
  webmin-slbackup
0 upgraded, 0 newly installed, 1 downgraded, 0 to remove and 0 not upgraded.
Inst webmin-slbackup [0.0.10-1] (0.0.9-1 Debian:3.1r0/stable)
Conf webmin-slbackup (0.0.9-1 Debian:3.1r0/stable)
tjener:~# apt-get install webmin-slbackup=0.0.9-1
Reading Package Lists... Done
Building Dependency Tree... Done
The following packages will be DOWNGRADED:
  webmin-slbackup
0 upgraded, 0 newly installed, 1 downgraded, 0 to remove and 0 not upgraded.
Need to get 22.0kB of archives.
After unpacking 131kB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://ftp.debian.org sarge/main webmin-slbackup 0.0.9-1 [22.0kB]
Fetched 22.0kB in 0s (23.6kB/s)
dpkg - warning: downgrading webmin-slbackup from 0.0.10-1 to 0.0.9-1.
(Reading database ... 32924 files and directories currently installed.)
Preparing to replace webmin-slbackup 0.0.10-1 (using .../webmin-slbackup_0.0.9-1_all.deb) ↩
    ...
Unpacking replacement webmin-slbackup ...
Setting up webmin-slbackup (0.0.9-1) ...
```

8.11 Install a package using dpkg

User case: Sometimes it is needed to download a package from other places, not located in a Debian web archive. Opera browser is such a package.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

Download the package from the home page of the creators of the program. This could for example be Opera. The program is installed using the following command:

dpkg -i <full path to the package>

. If you first want to simulate this try

dpkg --no-act -i <full path to package>

```
tjener:~# dpkg --install --no-act opera_8.51-20051114.5-sharedqt_en_sarge_i386.deb
Selecting previously deselected package opera.
(Reading database ... 32924 files and directories currently installed.)
Unpacking opera (from opera_8.51-20051114.5-shared-qt_en_sarge_i386.deb) ...
tjener:~# dpkg --install  opera_8.51-20051114.5-shared-qt_en_sarge_i386.deb
Selecting previously deselected package opera.
(Reading database ... 32924 files and directories currently installed.)
Unpacking opera (from opera_8.51-20051114.5-shared-qt_en_sarge_i386.deb) ...
dpkg: dependency problems prevent configuration of opera:
 opera depends on libqt3c102-mt; however:
  Package libqt3c102-mt is not installed.
dpkg: error processing opera (--install):
 dependency problems - leaving unconfigured
Errors were encountered while processing:
 opera
```

dpkg requires more manipulations than apt-get because it does not handle package dependencies. This means you may need to run **apt-get** immediately afterwards with additional parameter. For example, it helps to run **apt-get --fix-broken** to tidy up

```
tjener:~# apt-get install --fix-broken --simulate
Reading Package Lists... Done
Building Dependency Tree... Done
Correcting dependencies... Done
The following extra packages will be installed:
  libaudio2 liblcms1 libmng1 libqt3c102-mt libxcursor1 libxft2
Suggested packages:
  nas liblcms-utils libqt3c102-mt-psql libqt3c102-mt-mysql libqt3c102-mt-odbc
The following NEW packages will be installed:
  libaudio2 liblcms1 libmng1 libqt3c102-mt libxcursor1 libxft2
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
1 not fully installed or removed.
Inst libaudio2 (1.7-2 Debian:3.1r0/stable) [opera ]
Inst liblcms1 (1.13-1 Debian:3.1r0/stable) [opera ]
Inst libmng1 (1.0.8-1 Debian:3.1r0/stable) [opera ]
Inst libxcursor1 (1.1.3-1 Debian:3.1r0/stable) [opera ]
Inst libxft2 (2.1.7-1 Debian:3.1r0/stable) [opera ]
Inst libqt3c102-mt (3:3.3.4-3 Debian:3.1r0/stable)
Conf libaudio2 (1.7-2 Debian:3.1r0/stable)
Conf liblcms1 (1.13-1 Debian:3.1r0/stable)
Conf libmng1 (1.0.8-1 Debian:3.1r0/stable)
Conf libxcursor1 (1.1.3-1 Debian:3.1r0/stable)
Conf libxft2 (2.1.7-1 Debian:3.1r0/stable)
Conf libqt3c102-mt (3:3.3.4-3 Debian:3.1r0/stable)
Conf opera (8.51-20051114.5 )
tjener:~# apt-get install --fix-broken
Reading Package Lists... Done
Building Dependency Tree... Done
Correcting dependencies... Done
The following extra packages will be installed:
  libaudio2 liblcms1 libmng1 libqt3c102-mt libxcursor1 libxft2
Suggested packages:
  nas liblcms-utils libqt3c102-mt-psql libqt3c102-mt-mysql libqt3c102-mt-odbc
The following NEW packages will be installed:
  libaudio2 liblcms1 libmng1 libqt3c102-mt libxcursor1 libxft2
0 upgraded, 6 newly installed, 0 to remove and 0 not upgraded.
1 not fully installed or removed.
Need to get 3489kB of archives.
After unpacking 8753kB of additional disk space will be used.
Do you want to continue? [Y/n]
Get:1 http://ftp.debian.org sarge/main libaudio2 1.7-2 [71.5kB]
Get:2 http://ftp.debian.org sarge/main liblcms1 1.13-1 [123kB]
Get:3 http://ftp.debian.org sarge/main libmng1 1.0.8-1 [171kB]
Get:4 http://ftp.debian.org sarge/main libxcursor1 1.1.3-1 [23.7kB]
Get:5 http://ftp.debian.org sarge/main libxft2 2.1.7-1 [54.4kB]
Get:6 http://ftp.debian.org sarge/main libqt3c102-mt 3:3.3.4-3 [3045kB]
Fetched 3489kB in 16s (212kB/s)
Selecting previously deselected package libaudio2.
(Reading database ... 33027 files and directories currently installed.)
Unpacking libaudio2 (from .../libaudio2_1.7-2_i386.deb) ...
Selecting previously deselected package liblcms1.
Unpacking liblcms1 (from .../liblcms1_1.13-1_i386.deb) ...
Selecting previously deselected package libmng1.
Unpacking libmng1 (from .../libmng1_1.0.8-1_i386.deb) ...
Selecting previously deselected package libxcursor1.
Unpacking libxcursor1 (from .../libxcursor1_1.1.3-1_i386.deb) ...
Selecting previously deselected package libxft2.
Unpacking libxft2 (from .../libxft2_2.1.7-1_i386.deb) ...
```

```
Selecting previously deselected package libqt3c102-mt.
Unpacking libqt3c102-mt (from .../libqt3c102-mt_3%3a3.3.4-3_i386.deb) ...
Setting up libaudio2 (1.7-2) ...

Setting up liblcms1 (1.13-1) ...

Setting up libmng1 (1.0.8-1) ...

Setting up libxcursor1 (1.1.3-1) ...

Setting up libxft2 (2.1.7-1) ...

Setting up libqt3c102-mt (3.3.4-3) ...

Setting up opera (8.51-20051114.5) ...
```

Armed with different commands from earlier in this chapter, we can now confirm that Opera is already installed

```
tjener:~# apt-cache policy opera
opera:
  Installed: 8.51-20051114.5
  Candidate: 8.51-20051114.5
  Version Table:
 *** 8.51-20051114.5 0
        100 /var/lib/dpkg/status
tjener:~# dpkg --list|grep opera
```

ii opera 8.51-20051114. The Opera Web Browser

8.12 Search through files in a package

Use case: Want to find a program name or file in a package

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

You get an overview with the command

dpkg --listfiles <package name>

```
tjener:~# dpkg --listfiles opera
/usr/bin
/usr/bin/opera
.
.
.
/etc
/etc/opera6rc
/etc/opera6rc.fixed
```

8.13 Find which package a file came from

User case: Want to find the package a file came from.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

dpkg --search <filename>

This can look like this

```
tjener:~# dpkg --search /etc/opera6rc.fixed
opera: /etc/opera6rc.fixed
```

8.14 Unpackaging files from a package without installing the package

Use case: Perhaps an important system file was deleted by accident, and there is no backup.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

When using the command

dpkg --search <filename>

Warning: *Never* unpack packages in the root directory

if one finds which package a file came from. One can extract the package to get back the systemfile like shown later.

First you need to fetch the deb package in question. This can be done by placing it in the `/tmp` directory. Use this command to unpack the files in this directory

dpkg --vextract <package name> /tmp

. Then the required directories will be created in `/tmp` and the files are placed there.

dpkg --vextract <package name> /tmp

8.15 Make your own package mirror

Use case: Some packages are often installed. For others it is useful to avoid downloading them from the Internet.

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

The **apt-get** command makes it easy to install packages from the Internet. But **apt-get** will use significant network capacity when programs are downloaded from Debian archives on the Internet. Because of that, it is possible to tell **apt-get** to use a local package repository. This way it is possible to install already downloaded packages simply by using **apt-get**. This provides *quick* installation.

mkdir /var/www/dpkg

cp /var/cache/apt/archives/*.deb /var/www/dpkg

cd /var/www/

dpkg-scanpackages dpkg /dev/null | gzip -9c > dpkg/Packages.gz

After that, add a new line to the file `/etc/apt/sources.list`:

```
deb file:///var/www dpkg/
```

And then the command **apt-get update** must be executed as usual to update the packages in the database.

8.16 Secure login to the firewall (ssh)

Use case: Some times it is required to log into Coyote Linux when no web browser is available. Perhaps the command line is preferred? Then ssh can be used to connect to Coyote Linux.

If you are logged into a machine in a Skolelinux / Debian Edu network you can use

ssh -l root 10.0.2.1

to log in on Coyote Linux

If you are outside a Skolelinux / Debian Edu network, the value 10.0.2.1 can be replaced with the appropriate value for the network card with the WAN in. In this case it might be **ssh -l root 192.168.1.10**

Here you will meet the same options present as when logged into the Coyote Linux web administration. This is presented in a text based menu.

```
               Coyote Linux Gateway -- Configuration Menu


  1) Edit main configuration file          2) Change system password
  3) Edit rc.local script file             4) Custom firewall rules file
  5) Edit firewall configuration           6) Edit port forward configuration

  c) Show running configuration            f) Reload firewall
  r) Reboot system                         w) Write configuration to disk

  q) quit                                  e) Exit
  -----------------------------------------------------------------------------
  Selection:
```

The options will be approximately the same as those provided when logged into Coyote Linux for web administration. See section 3.7 for a quick description of the menu choices.

When selecting **q) quit** you will end up on the command line in Coyote Linux. If you need to get back to the main menu in Coyote Linux, write **menu** and press **Enter**.

If you see this when you try to log into Coyote Linux

```
klaus@tjener:~$ ssh 10.0.2.1 -l root
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
@    WARNING: REMOTE HOST IDENTIFICATION HAS CHANGED!     @
@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@@
IT IS POSSIBLE THAT SOMEONE IS DOING SOMETHING NASTY!
Someone could be eavesdropping on you right now (man-in-the-middle attack)!
It is also possible that the RSA host key has just been changed.
The fingerprint for the RSA key sent by the remote host is
34:b7:a3:9b:06:4c:e2:30:1b:0d:03:45:7b:22:b7:dd.
Please contact your system administrator.
Add correct host key in /skole/tjener/home0/klaus/.ssh/known_hosts to get rid of this ↩
    message.
Offending key in /skole/tjener/home0/klaus/.ssh/known_hosts:27
RSA host key for 10.0.2.1 has changed and you have requested strict checking.
Host key verification failed.
```

This is most likely because one logged in earlier into another machine with the IP address 10.0.2.1, or because the network card in Coyote Linux has been changed. It could also be an attack from an unknown man in the middle. The solution is to remove the key, in this case line number 27 in the `/skole/tjener/home0/klaus/.ssh/known_hosts` file.

8.16.1 Exception handling

8.16.2 Verification

8.16.3 Update the configuration database

8.17 Status summary for the firewall (Coyote)

Use case: Which commands can be used to get the menu or to get an overview of the status of the firewall?

Main author: Klaus Ade Johnstad

Useful commands in Coyote Linux

- ping

Useful to figure out if the network is working. The command checks if there is a connection to the Skolelinux / Debian Edu main server.

```
coyote# ping -c5 10.0.2.2
PING 10.0.2.2 (10.0.2.2): 56 data bytes
64 bytes from 10.0.2.2: icmp_seq=0 ttl=64 time=0.9 ms
64 bytes from 10.0.2.2: icmp_seq=1 ttl=64 time=0.5 ms
```

- uptime

This command gives the duration since the last reboot for Coyote Linux.

```
   coyote# uptime\n  2:37pm  up 80 days,  7:55, load average: 0.00, 0.00, 0.00
```

- dmesg

This command displays information about the Linux kernel running on the machine. It lists things like memory, processor and network card. If there is too much output from **dmesg** you can send the output through a so called pager program like "more" and use **Space** to read everything, **dmesg|more**

- ifconfig

Show extra information about the network cards.

```
coyote# ifconfig
eth0      Link encap:Ethernet  HWaddr 00:50:FC:F8:D2:44
          inet addr:10.0.2.1  Bcast:10.0.3.255  Mask:255.255.254.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:314723 errors:0 dropped:0 overruns:0 frame:0
          TX packets:312105 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:53700845 (51.2 !MiB)  TX bytes:277496136 (264.6 !MiB)
          Interrupt:11 Base address:0x7000

eth1      Link encap:Ethernet  HWaddr 00:E0:18:A8:B1:BA
          inet addr:192.168.100.133  Bcast:192.168.100.255  Mask:255.255.255.0
          UP BROADCAST RUNNING MULTICAST  MTU:1500  Metric:1
          RX packets:307395 errors:0 dropped:0 overruns:0 frame:0
          TX packets:281202 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:1000
          RX bytes:272404311 (259.7 !MiB)  TX bytes:47880640 (45.6 !MiB)
          Interrupt:10 Base address:0xb800 Memory:e3000000-e3000038

lo        Link encap:Local Loopback
          inet addr:127.0.0.1  Mask:255.0.0.0
          UP LOOPBACK RUNNING  MTU:16436  Metric:1
          RX packets:14565 errors:0 dropped:0 overruns:0 frame:0
          TX packets:14565 errors:0 dropped:0 overruns:0 carrier:0
          collisions:0 txqueuelen:0
          RX bytes:1290756 (1.2 !MiB)  TX bytes:1290756 (1.2 !MiB)
```

- lsmod

This command lists up driver modules. This is useful to see which modules are used by network cards.

```
coyote# lsmod
Module                  Size  Used by
eepro100               17516   1
3c59x                  24408   1
mii                     1852   0 [eepro100]
ip_nat_quake3           1608   0 (unused)
ip_nat_mms              2448   0 (unused)
ip_nat_h323             2044   0 (unused)
ip_nat_amanda           1020   0 (unused)
```

This is a list showing that the driver modules for the network card are loaded. For Intel pro100 the module is named eepro100 and for 3Com the module is named 3c59x (which is valid for cards with type names 3c590, 3c595, 3c900, 3c905). See section 3.12

- route
- traceroute

Is useful to figure out where the Internet packages move. If there are problems, it is useful to see the path the Internet packages use.

- showcfg

Another command giving information about the state of the network cards.

```
Coyote running configuration display utility.

Internet    (eth1): UP
LAN network (eth0): UP

-------------Internet configuration--------------
IP Address   192.168.100.133 (Static)
Netmask      255.255.255.0
Gateway      192.168.100.2
---------------LAN configuration----------------
IP Address   10.0.2.1
Netmask      255.255.254.0
Broadcast    10.0.3.255
---------------DNS configuration----------------
domain localdomain
nameserver 213.184.200.1
nameserver 213.184.200.2
------------------------------------------------
10:51am up 7 days, 20:53, load average: 0.00, 0.00, 0.00

Press enter to return to system menu.
```

- free

The command is used to see how much memory is available and how much is used. This machine has 32 MB memory.

```
coyote# free
             total         used         free       shared      buffers
  Mem:        30860         6004        24856            0            0
 Swap:            0            0            0
Total:        30860         6004        24856
```

- menu

This command starts the Coyote Linux menu

```
              Coyote Linux Gateway -- Configuration Menu


 1) Edit main configuration file         2) Change system password
 3) Edit rc.local script file            4) Custom firewall rules file
 5) Edit firewall configuration          6) Edit port forward configuration

 c) Show running configuration           f) Reload firewall
 r) Reboot system                        w) Write configuration to disk
```

- reboot

coyote#reboot

This command does a reboot of Coyote Linux

- shutdown

coyote#halt

Here is Coyote Linux turned off

8.18 Next

Use case:

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

8.18.1 Exception handling

8.18.2 Verification

8.18.3 Update the configuration database

8.19 Last

Use case:

Author: Klaus Ade Johnstad.

Co-author: Knut Yrvin

8.19.1 Exception handling

8.19.2 Verification

8.19.3 Update the configuration database

9 Copyright and authors

This document is written and copyrighted by Knut Yrvin (2006), Andreas Johansen (2006), Klaus Ade Johnstad (2006), Halvor Dahl (2006), Snorre Løvås (2006), Finn-Arne Johansen (2006), Ragnar Wissløff (2006), Petter Reinholdtsen (2015),

Ole-Erik Yrvin (2015) and Ingrid Yrvin (2015) and is released under the GPL3 or any later version. The source document was originally written in Norwegian Bokmål, and migrated to English in 2015.

If you add content to it, **please only do so if you are the author. You need to release it under the same conditions**! Then add your name here and release it under the "GPL v3 or any later version" licence.

10 Appendix A - Contract on operating Debian Edu / Skolelinux

Contract no.: .....

Customer no.: .....

10.1 CONTRACT ON OPERATING DEBIAN EDU / SKOLELINUX

between

Driftselskapet AS, Maskinrommet 1, 0313 Oslo

Org.no.: 989 313 313

(hereafter called The Vendor)

and

NN

Org.No:

(hereafter called The Customer)

The parties have reached an agreement on the delivery of operational services (hereinafter The Agreement) on subsequent contractual terms. The following appendixes are part of The Agreement:

- Appendix 1 - Definitions
- Appendix 2 - Customer Obligations
- Appendix 3 - The Vendor's obligations
- Appendix 4 - Prices and terms of payment
- Appendix 5 - General provisions
- Appendix 6 - The proxy Persons

The agreement is valid from the signing date and a minimum of 12 months from The Delivery date. The agreement is then renewed automatically for periods lasting 12 months unless one of the parties denounces the Agreement in writing, three months before the expiry of a contract period.

The contract is signed in two - 2 - copies, and each of the parties keeps one - 1 - copy.

Place: .....

Date: 2006

For The Vendor: .....

For The Customer: .....

10.1.1 Appendix 1 - Definitions

Term	Description
Operating period	From the Delivery day to the day when the agreement ceases to apply, regardless of the reason.
Services provided	Services from The Vendor in the Operating period. The services provided are further described in Appendix 3.
ICT manager	Competence person(s) at the customer serving as liaison(s) to the supplier.
Delivery day	The day the customer can use the services provided.
Skolelinux	Linux distribution built on Debian Linux and adjusted for use in Norwegian schools.

10.1.2 Appendix 2 - Customer Obligations

10.1.2.1 1. ICT skill requirements

ICT administrator (1 - 3 named persons at the customer) to deal with inquiries from users related to the use of the applications included in Skolelinux/Debian Edu. ICT administrator shall have sufficient expertise to make a qualified assessment of whether a problem is related to the use or operation of the system.

The ICT administrator should contact the supplier's the user support center by phone or e-mail. The customer's users should not contact the supplier directly.

10.1.2.2 2. Machine requirements

The Customer should have installed and tested that the equipment operates satisfactorily before the delivery day.

10.1.2.3 3. Program requirements

The customer shall, before delivery day, have installed Skolelinux / Debian Edu to get a verified, satisfactory functioning installation.

10.1.2.4 4. Communication requirements

The Customer shall, before the delivery date, have installed and configured communication with the Internet and tested it works satisfactorily. To make it possible to provide services, the customer must arrange for the contractor to be able to access the customer's ICT-facilities via the Internet.

10.1.2.5 5. Information from The Vendor

When all the above requirements are met, the customer shall notify the contractor, in writing or by e-mail, that the ICT-system is prepared for the contractor for provide services.

A list of all the users of the system including full name, username and wanted password should be sent electronically to the Vendor at the latest together with this message.

10.1.3 Appendix 3 - The Vendor's obligations

10.1.3.1 1. Delivery day requirements

The supplier shall, after receiving notification from the customer in accordance with Appendix 2, paragraph 5, as soon as possible arrange for the Customer to receive the provided services. Delivery date shall be no later than 4 weeks after such notice is received by the supplier.

10.1.3.2 2. Information to The Customer

When all the above requirements are met, the contractor shall notify the customer, in writing or by e-mail, that the ICT-system is prepared for the customer to receive the provided services.

10.1.3.3 3. Service requirements

The following table shows all relevant services related to operating Skolelinux/!DebianEdu. The crosses in the table show the responsibilities between the Supplier and the Customer for the different services:

Delivered (incl.) are carried out by the supplier and included in the Agreement price. Delivered (running) performed by the Supplier at the Customer's account in accordance with the rates in Chapter 7. The Customer, is done by the Supplier at the Customer's expense.

Service	Delivered (incl.)	Delivered (running)	The Customer
Troubleshooting and user support over the phone and email	x		
Participation in the user forum	x		
Replacing the hardware[1]	x		
Add, change and remove users[2]	(x)		x
Changing password when the password is forgotten	(x)		x
Security updates on Skolelinux	x		
Version updates on Skolelinux	x		
Change the user permissions	(x)		x
Monitoring of filling on disks	x		
Monitoring of the lifetime for the relevant components	x		
Extending disk partitions	x		
Operation and monitoring of firewall	x		
Operation and monitoring of network	x		
Deleting print jobs stuck in the queue at the request of the ICT administrator	x		
Monitoring to ensure backup copies are taken	x		
Data deletion under request from the ICT administrator	x		
Replacing backup medium and storing backup copies	x		
Restore with a security backup, at the request of the ICT administrator.	x		
Set up new printers and printer queues	(x)		x

Stopping and restarting the printer queues at the request of the ICT administrator	x	
Stopping hanging processes on the server as a result of application errors	x	

10.1.3.4 4. Response time requirements

The supplier shall without undue delay, start troubleshooting and problem solving. ICT administrator should be held continuously updated on the status and progress of error correction.

10.1.3.5 5. Skill requirements

The supplier shall at all times have sufficient resources with relevant expertise to provide services in a professional manner

10.1.4 Appendix 4 - Prices and terms of payment

10.1.4.1 1. Compensation for services provided

The compensation for the services provided is calculated on the basis of the number of workstations on the network. The agreement includes a minimum of 60 workstations. The Customer shall pay the Supplier £78 per year per workstation, excluding VAT, in compensation for the services provided, i.e. £390 per month excluding VAT for 60 workstations.

If the number of workstations changes the customer shall give the supplier a written notice thereof with the corresponding dates for the change. Adjustment of the billing basis with a possible recalculation will be included in the next invoice

10.1.4.2 2. Consultant support

Hourly rate for consultancy is NOK 800 (65 £) ex Moms (VAT). All work on an ongoing bill should be approved by the customer before work starts. Documented travel expenses are charged to the client. Compensation for travel time calculated by the elapsed time with hourly rate NOK 400 ex Moms.

10.1.4.3 3. Payment conditions

Compensation for the services provided is billed in advance for each quarter. For the first quarter, billing starts from the delivery date and runs until the end of the current quarter.

Compensation for consultancy is billed as after-payment on the basis of agreed and work performed.

All invoicing is done within a 30 days deadline.

10.1.4.4 4. Price regulation

Prices may be adjusted every year with the increase in the national consumer price index (SSB CPI). This can take place for the first time one year after signing the agreement.

[1] Supplier's responsibility is limited to managing the a change of hardware. The supplier is not responsible for hardware and warranties, pricing, shipping costs etc. which must bee agreed separately with machine supplier.

[2] The customer can do this using a separate application in Debian Edu. The supplier can do this server for NOK 50 per user excluding vat.

10.1.5 Appendix 5 - General provisions

10.1.5.1 1. The parts' cooperation and duties

General

The parties shall cooperate to achieve the most efficient implementation of the Agreement. Both parties may, in writing, summon one another to meet with five business days' notice to discuss matters arising in connection with the implementation of the Agreement. The parties are obliged, without delay, to notify each other about matters that they understand or should understand may affect the implementation of the Agreement. Such notification does not relieve the parties from the responsibilities resulting from the Agreement.

The suppliers duties

The Supplier undertakes to supply the contract business performance at the terms of the Agreement. The supplier undertakes to allocate the resources necessary to implement the commitments in the Agreement.

Customer duties

The customer shall pay the agreed compensation. The customer must assist the supplier so that the supplier will not be delayed or otherwise prevented from fulfilling the obligations. The customer undertakes to allocate the necessary resources, and ensure the necessary assistance from a third party where this is agreed.

10.1.5.2 2.Confidentiality

The parties are mutually obliged to keep confidentiality and not disseminate information which they become aware of in connection with carrying out the out the Agreement, to the extent that such information is not considered public. The same applies to all the material which is marked confidential. personal matters, and information that could harm the parties or that can be exploited by outsiders in business. This duty of confidentiality applies to the parties and their employees and others acting on behalf of the parties in connection with carrying out the of the Agreement. The duty of confidentiality applies correspondingly after the termination of the Agreement.

10.1.5.3 3.Force majeure

In the event of an extraordinary situation outside control of the parties, which could not be foreseen at inception and which significantly hampers the fulfilment of a party duties, the other party shall be notified without undue delay. The affected party's obligations are suspended to the extent that is relevant so long as the extraordinary situation prevails. The other party in return suspended for the same period. Either party may terminate the Agreement by giving one month's written notice if the force majeure situation makes it particularly burdensome to maintain the Agreement.

10.1.5.4 4. Transfer of the agreement

Parties may only reassign their rights and obligations under the agreement with the written consent of the counterparties. Consent may not be unreasonably withheld. It is not considered as transfer if one of the parties mergea with one or more other companies or the assignment is to a subsidiary. Right to compensation under this Agreement may be assigned freely, but such transfer does not relieve the Contractor from its obligations and responsibilities.

10.1.5.5 5. Non-fulfilment

10.1.5.5.1 5.1 Delay of the delivery date

10.1.5.5.2 a. Liquidated damages

If delivery does not happen on the date agreed between the parties, and this is not due to the circumstances mentioned in Clause 3 or circumstances the Customer is responsible for, then a daily penalty is applied from the agreed delivery date. The penalty fee is 0.1% of the agreed annual compensation for the portion of services provided that are delayed, calculated per calendar day of delay and up to a maximum of 60 days. As long as daily penalties are being applied, the customer may neither terminate the Agreement nor demand a discount or other compensation for the delay.

10.1.5.5.3 b. Canceling

If the delivery date has not occurred by the end of liquidated damages period, you may terminate the Agreement with immediate effect.

10.1.5.5.4 c. Delay caused by customer

In case of delay caused by customer the supplier may, by written notice, cancel their work until the customer takes corrective action. The supplier is entitled to recover their additional costs as a result of customer's breach, and a reasonable time to the reassignment of resources.

10.1.5.5.5 5.2 Defaults in the operating period

10.1.5.5.6 5.2.1 The suppliers non-fullfillment

10.1.5.5.7 a. Shortcomings

There is a shortcoming of the supplier if services provided do not meet the requirements and specifications stipulated in the Agreement, and this causes a circumstance for which the supplier is responsible. If there is a shortcoming in operating performance, the supplier shall without undue delay remedy the defect. Where a defect can not be repaired within a reasonable time the Customer shall be entitled to a proportionate discount, ref. Section b. Below.

10.1.5.5.8 b. Price discount for shortcomings

If the client has not been able to use the services provided, fully or partially, as a result of the defect, the customer has the right, in the period from when the error or defect was notified in writing until the defect is corrected, to receive a proportionate discount. Any refund due to lack of availability due to the same circumstance, is deducted when calculating the discount.

10.1.5.5.9 c. Canceling

In the event of any other shortcoming, that is significant to the customer's use of the services provided, and is not corrected within 30 business days after the Customer notified the supplier in writing about the shortcoming, the Customer may notify the supplier in writing of intent to terminate the Agreement. If the supplier, after such notice, has not rectified the situation within 14 business days, the customer is entitled to terminate the Agreement with immediate effect.

10.1.5.5.10 5.2.2 The customers non-fullfillment

If the customer does not pay on time, the supplier is entitled to interest for the amount that is overdue. (In Norway, this arises from a law relating to interest on late payment of 19. Dec. 1976 no. 100, § 3, first paragraph.) In cases where the payments plus interest are not paid within 14 days of the due date, the supplier can issue written notice that the services provided will be discontinued, or that the agreement will be terminated, unless the customer settles all outstanding bills within 7 days of receipt of this notification. Upon termination of the Agreement due to the customer's fault, the supplier shall be indemnified by the customer for the costs and liabilities undertaken in connection with the Agreement.

10.1.5.6 6.Replacement

The customer may demand compensation for losses that can be reasonably attributed to the shortcoming, unless the supplier can demonstrate that the breach, or the cause of the breach, not attributed to him. Any liquidated damages caused by delay in accordance with Clause 5a for the same breach is deducted by calculating compensation. If the customer defaults on its obligations under this Agreement, supplier shall be entitled to recover their additional costs that may reasonably be attributed to the Customer defaults, unless the Customer can prove that the breach, or cause of the breach can not be attributed to him.

Parties are not responsible for the other party's indirect losses, including expected savings or gains. Indirect losses included among others

- Losses due to reduced or lost production or sales (operational interruption);
- Losses arising because the services provided can not be used as intended (consequential losses);
- Lost profits as a result of a contract with a third party that is dropped or not fulfilled properly.

Parties liability towards each other is limited to the agreed annual compensation, or a maximum of NOK 1 million, regardless of the number of damage cases. The limitations of the parties' liability does not apply, if the party or anyone he is responsible for, has shown gross negligence or willful misconduct.

10.1.5.7 7. Legal defects

If a third party asserts that the use of software that the Customer or Vendor has license responsibility goes against the third party's rights, the Party shall ensure that appropriate rights are retained or acquired, or that other equivalent software functionality is obtained without charge to the other party. Should claims arise from a third party against the Customer or Vendor on the basis of defects inherent in the relationship of the other Party, that Party undertakes its own expense to assist and eventually lead case for both parties. From the time a party takes over the case, the other party is obliged to assist the special compensation.

10.1.5.8 8. Responsibility for subcontractors

Parties are fully accountable for agreed services that are performed by subcontractors.

10.1.5.9 9. Regulating the termination of the Agreement

Upon termination, the parties shall draw up a joint plan of liquidation of the customer relationship and obligations by mutual to assist each other in the practical work in this liquidation. The vendor is obliged by termination of this Agreement to return Client software and current data in the agreed format. The Customer chooses the means of transport and is responsible for transportation from the Vendors' premises. The Customer undertakes immediately after termination of the Agreement to return all equipment belonging to the Vendor. The Vendor chooses the mode of transport and is responsible for transportation from the Vendor's premises.

10.1.5.10 10. Legalities and solving disagreements

The rights and obligations under this Agreement shall completely follow the Norwegian law. Upcoming Disagreements in connection with this Agreement shall be resolved by negotiation between the parties. If the parties fail within two weeks not to solve the disagreement through negotiations, either party may require the dispute to be resolved by arbitration under the rules of the law of 13 August 1915 No. 6, Chap. 32 (Civil Procedure). Each party shall appoint one arbitrator who together appoint the arbitration tribunal. If a party fails to designate its representative within two weeks after the other has demanded arbitration and appointed its representative, he will be appointed by the Chief Justice of the Oslo District Court. The same applies for the election of the chairman if the two arbitrators members have not chosen the President within 14 days after both being appointed.

10.1.6 Appendix 6 - Contacts and addresses

1. Correspondence Requests regarding the agreement shall be in writing and addressed as follows:

To Vendor	To Customer
Operating company LtdBy authorized personMachine room 10313 Oslo	NNBy authorized person

2. Authorized persons

The following persons do have authority to sign on their part according to the agreement.

Name	Position/Function	Telephone	Telefax	E-mail
The vendor				
Petter Smart	CEO	+47 22 31 31 31	ps@driftselskapet.no	
The Customer				