Master Education Programmes in Network and System Administration

Mark Burgess – Oslo University College
Karst Koymans – Universiteit van Amsterdam

ABSTRACT

We report on and discuss our experiences with teaching Network and System Administration at the level of Masters at Oslo University College and the University of Amsterdam. At our respective institutions we have independently arrived at very similar models for teaching a traditionally vocational subject within an academic Computer Science framework by incorporating a strong practical component.

Introduction

For several years now academically inclined system administrators have struggled to identify the role and place of System Administration within the fields of Computer Science and Engineering. This effort has often brought controversy, with strong and diverging opinions dominating over any consensus. This is not uncommon for a novel field of study. Whereas many university subjects evolve from academic and vocational traditions that have existed for generations, the establishment of a curriculum in System Administration (a hybrid subject somewhere between computing, science and engineering) cannot make the progress industry and society needs by waiting for the results of a protracted evolutionary process. The need for system administrators is here now.

In our degree programmes we have chosen to avoid the controversies and formulate courses based on our own experiences as system administrators and academics. We believe that professional educators have a better chance of settling these controversies than those whose passions rage, as we shall explain below. By anchoring the subject in established academic traditions, but maintaining the hands-on aspect of the subject, university milieux have been more receptive to the idea of system administration as an academic discipline, even when they have not always understood the initial vision. What is interesting is that, in spite of the lack of standardization in thinking, and in spite of superficial practical differences imposed by our local environments, the courses we have developed in Oslo and Amsterdam overlap strongly in both flavour and substance, so perhaps consensus is not so far from reality after all.

Our aim in this paper is to report on our efforts in this area. We do not present our courses as perfectly formed, finished products to be admired by all (no university course ever turns out the way its designers would like, due to numerous constraints and obstacles), rather we comment on the philosophies and implementations that have led us to make courses at our Universities. In each case, although we each have some relevant introductory Bachelor level courses, we have found that Masters level study programmes are most suitable for implementing system administration studies, since students benefit from Bachelor level skills in more standard computer science as well as from a certain breadth of background. We shall not attempt to view system administration as a profession in the sense of apparently similar organized job descriptions (like the medical profession), as this topic is charged with issues that go far beyond education. Rather we focus on our experiences as educators and point out how we have approached a problem that some have claimed was impossible: to turn system administration into a discipline.

We begin by discussing our approaches to educating system administrators, then we consider norms and standards in related fields. We summarize the content that we consider to be essential and report on our experiences with this. Finally we draw some initial conclusions about our successes and failures. We hope that the strong similarities we have arrived at in our own neutral attempts to formulate system administration academically will help others to see their own subject or profession through more impartial eyes, and perhaps even help to advance the state of understanding of the field.

Relationship of System Administration to Computer Science

Let us begin by asking how system administration relates to other fields of research that are commonly associated with it. Computer Science derives traditionally from two camps: mathematical logic and electrical engineering. Both of these have long academic traditions that have influenced the way computing has been researched, framed and is taught in Universities. In addition it is known that physics graduates are well represented in some areas of Computer Science. Not everyone in working with computers has learned the subject in a university. Computing as a phenomenon is young and many self-taught hobbyists have entered the workplace on the strength of their own private initiative. Such people ought not feel threatened or belittled by academic initiatives.

System administration has not normally been a subject in its own right, though many short courses have been offered by universities around the world. However it has not been completely absent from curricula. Some work that has gone into studying "computer management" from universities and graduates has been in the area of telecommunications, thanks to often generous sponsorship of powerful telecom organizations. Much of this work has been closed-source however and centered around organizations like the Telemanagement Forum (TMF) and the Internet Engineering Taskforce (IETF). It goes back to the 1970s under the title of "Network Management" and, in some respects, many of the issues facing system administration have been discussed and "solved" within that limited context, e.g., see [1]. Thus, system administration lags behind its related "big brother" Network Management. Many electrical engineers who have found themselves in the clutches of information technology revolution have entered this field through telecommunications. It is taught in various courses especially in Europe (where most of the research in this field is carried out) and it is represented by major conferences like NOMS [2] and IM [3] and the IARIA [4] conferences, organized by major telecommunications and routing companies like Cisco and Motorola. Since about 2001 we have made a concerted effort to cross-pollinate these disparate communities.

A second group that has involved itself in management concerns is software engineers. Distributed software systems and middleware are often used to "manage" software layers, instrumenting software with inter-communication capabilities that need to be managed just like more complete operational environments. Although the list of challenges is somewhat smaller in software engineering, and this gives software engineers an oversimplified impression of the challenges of system administration, the overlap makes a connection with software engineering. It also contributes to the widespread belief amongst computer scientists that system administration can be solved through software engineering alone, though this seems to be changing as computer systems become more ubiquitous (see conferences like DSOM [5] for these communities).

How Should System Administration Be Taught?

The need for formal education is rarely disputed but the form is often controversial, especially amongst those who learned some of the necessary skills flying by the seat of their pants. It is common for self-taught practitioners to reduce subjects to a list of a few skills that are needed, then "the rest is experience." It has even been suggested that it is "too early" to formalize anything about the field, since it is not yet well enough understood. However, the job of a skilled educator is to pragmatically distill such experience into a literature of material that can be taught, and we believe that it is never too soon to do this. Sufficient understanding is a journey rather than a finished product.

There is no single approach to learning that can solve all of a society's needs. The need for learning does not disappear once we are finished with school and so there is a need to combine further education with work in some appropriate way. As we shall discuss below, society is becoming less tolerant of paying for schooling, and is increasingly pressuring both enrolled students and would-be students to work alongside their further education.

Education is typically covered by a three-pronged strategy which involves:

- Self Learning.
- Training.
- College/university studies.

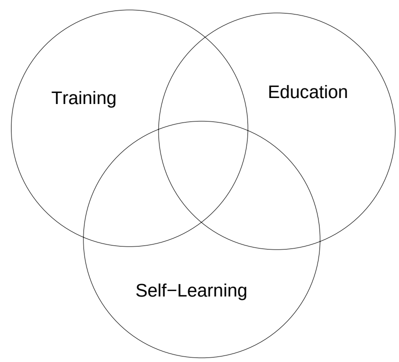

Figure 1: The three aspects of learning.

Although most education in system administration is in the first two categories, it was our intention to formalize the subject into an academic discipline in our master degree programmes. This has been a considerable challenge and has taken the better part of ten years of preparation in both our groups.

Our colleges have contributed several important text books in this field, one at Bachelor/Master level [6] and one at Masters/Ph.D. level [7]. A book for bachelor level studies also covers the Norwegian market [8].

Self-Learning

Self-learning is learning without the guidance of a tutor or course coordinator. There is no assessment and no interactive diagnosis of a student's progress. Self-learning is clearly a necessary part of any learning scheme learning without some effort by oneself would require technology yet to be invented (some would say that teaching is, in fact only management, that all we do is promote an environment which accelerates the student's own learning process by providing for, inspiring them and guiding them). Self-learning however implies that a student does not have access to an organized programme of study and is therefore lacking in the potential benefits of others' experience.

Self-learning is usually the only option available to the would-be student at the outset of a new field of study. In engineering disciplines particularly it remains an important aspect of a learning strategy, in the guise of "trial and error" practice. Getting one's hands dirty in the field is one of the most important confidence building exercises, and it is a fast track to connecting experience to meaning. Trial and error is an efficient approach to knowledge acquisition because one can easily see the patterns of success and failure at first hand. Both authors have had their share of self-learning experience.

Self-learning is not the same as experimentation or lab work. Many technicians working in laboratories are simply carrying out practices procedures that are often later replaced by automated machines. There does not need to be understanding to complete a task. Self-learning implies a development of understanding through one's own effort. The effort might simply involve reading, but in most cases it requires a person to engage their motor functions and do something physical (whether the doing is simply writing, solving math problems, or plugging cables into boxes).

We might define a subset of self-learning to be self-training, which we understand as the study of recipe-solutions to problems from standard materials. This is a common form of training for certification programmes. Self-learning can therefore span both education and training, in the following sense: significant reading can lead to a broad perspective on problems if the reading is broad enough and the self-taught student has the opportunity to put the reading into context.

Training

Training is a form of targeted knowledge presentation often aimed at teaching skills or giving summary overviews. It can never be a substitute for a complete education; it is mainly useful for filling in gaps in a basic knowledge base, such as learning a particular procedure or using a new tool. Training is not a realistic strategy for teaching complete and 'rounded' professionals because there is inevitably a need for long-term integration of knowledge into one's existing cultural base.

For people in full time employment, training is often the only available alternative to interact with a teacher. Short courses given at conferences or by other providers offer an access route to organized classes when students have time limitations, such as during full-time employment. Training courses are usually based on a set of slides and are presented at a low level for a wide audience, since they are financed by their commercial success. This leads to a side-effect of training which is an unintentional "dumbing-down" of the material in order to reach the largest paying denominator.

Some companies offer training programmes to graduates because they see gaps in their knowledge in specific skill areas. A graduate already has a basis of deep knowledge into which a training course can be integrated. A notable training programme that exceeds most is the Cisco Academy/university training programme.

The traditional form of training is that of half-day to one-day classes at conferences such as LISA, USENIX, NOMS [2], IM [3], etc. These are slide shows with commentary. Training tends to be supplied as a list of recipes and do's and don'ts. There is little time to develop any significant understanding. If an idea does not resonate with the trainee more or less immediately, he or she has little recourse other than recommended texts to fill in the gaps.

Training can be a "leg up" to help a motivated employee to self-learn for instance. This motivational aspect of teaching should not be underestimated. Training at conferences forms part of a (hopefully) positive experience that can have a powerful motivational effect. It is also sometimes accompanied by certification to capture part of the benefits of verification. Certification involves some kind of test, normally a multiple choice pychometric evaluation. Verification and testing can be employed in principle to ensure that trainees reach a standard interpretation of the subject of training, rather than their own (perhaps distorted) version.

College Studies

Accredited college or university education is the oldest kind of higher education, the most extensive and the only kind of learning that addresses all of the strategies for learning at the same time. A planned educational programme in a college or university has the potential to incorporate elements of both of the foregoing approaches within an extended curriculum. The ability to test students repeatedly serves also the purpose of measuring their abilities and motivating them to improve their performance.

A college education can build up concepts and principles over an extended period of time, and place them in a larger context. This is an important cultural experience. Humans learn essentially by story-telling, in which they must form their own version of a story, and place themselves within it, before they are willing to believe in it fully. An extended programme can explore with students the reasons behind what they are learning and build their own personal motivation. In a college programme, students have time for trial and error and they have the opportunity to put knowledge into a context.

College education is more interactive than either training or self-learning. There is a greater opportunity for feedback, building confidence and quickly correcting misconceptions. Moreover, the experience has an epidemic effect – one conversation with a student can improve the teacher's understanding and spread to another students, and so on in a viral way.

Another phenomenon introduced by college education is the concept of ''group self-learning,'' where students study in the alternating role of teacher and trainee. This so-called ''power learning'' is a useful extension to the materials and wisdom brought in by the official teacher. Both our institutions have practiced this with success.

Pre-requisites and the Culture of Learning

So what ought students know before and after their education? As university lecturers, we affirm that the issue of what students ought to know in advance of a course programme is highly politically charged. Education is a cultural phenomenon and priorities are constantly changing (not always for the better). Colleges and universities inherit students from other schools and workplaces, and cannot guarantee that applicants will reach the minimum bar expected for starting.

One difference between our study programmes lies in the attention to pre-requisites. At Amsterdam, far more work is put into selecting strong applicants and deterring weaker candidates than at Oslo, because the curriculum is taught in half the time and under greater pressure. An additional year for personal growth allows the course at Oslo to pull through students who might not make it in the pressure-cooker model used at Amsterdam.

Both institutions require basic programming, knowledge of operating systems principles and some Bachelor level mathematics. No institution can ever teach everything someone might need to know about a subject and thus colleges and universities do not try to do so. They do not usually give ''training'' in specific skills except as an example to a more general discussion. Rather, the approach is to charge students with the more fundamental skills needed to learn for themselves, along with a critical eye so as to not take everything at face value. Educational institutions have developed strategies for teaching these general skills over centuries. These methods have become cultural norms, and there are many reasons why we should respect them as norms, even if they do not train students to specifically use tool X or carry out procedure Y.

Teaching educational norms is important especially because it results in graduates who can communicate in a cross-disciplinary field and society at large. Examples of teaching norms include basic science and language skills (reading, comprehension, reporting, creative writing, summary etc.). Mathematics and physics are taught, for example, not usually to allow people to calculate planetary orbits, but because the skills one must surmount to complete these topics have general validity. Moreover, since most other people have been through the same experience, it becomes shared knowledge and this allows any student to communicate with any other person who has learned the same set of standard concepts. (''Remember how we used to calculate the efficiency in physics? We could do the same thing here...'') This would not be possible if the training were too specific and too directed. The ability to write clearly and to reason about problems is something else students of mathematics and physics learn. Writing and language skills have many of the same features of mathematics: grammatical structure, attention to detail, the need to interpret the meaning from a potentially ambiguous signal, and so on.

This cultural aspect of education is under pressure today from impatient and under-educated spokespersons in our societies who would dumb it down. The urgent rush to specialize and ''train'' people for ''the jobs industry needs today'' eventually becomes harmful to the culture of learning in society. If we bypass well-known metaphors in favour of new short-cuts, sometimes using computer software to remove the need even for a basic skill to be learned, then we impoverish the experience. We need these cultural norms in education, like math and science, both because they allow us to speak a *lingua franca* of reason that crosses discipline boundaries, and also because they allow us to re-use the experiences and hard-won understandings of other fields.

We believe it is important to understand the limitations of basic training: a trainee can repeatedly complete a task without any understanding whatsoever. It is only when the train runs off its tracks somewhere that one discovers that the uneducated student has no idea where he or she is in the landscape of knowledge.

And this is exactly where the need for academic level education comes in. Not only to lay out the tracks and keep the train on those tracks (which can be trained more or less in the above mentioned education shortcut), but also in the case of derailment or when entering unknown territory.

Modeling

The philosophy of science gives us one of the most essential tools in the problem solver's toolkit: the idea of modeling. As the philosopher of science David Hume maintained, there are two kinds of knowledge that should be clearly distinguished.

- Theoretical knowledge that can be determined precisely and exactly (proven perhaps), but whose relationship to the real world is uncertain.
- Empirical knowledge that is certainly about the real world, but whose measurement and interpretation are not exactly determined.

In both cases our understanding of the world is imperfect. Our basic understanding of phenomena involves building approximate mental models, so a deeper understanding of this process can only help students towards a deeper understanding of the phenomena also.

A model is composed of suitably idealized approximations that attempt to manage this uncertainty and allow one to:

- Make predictions.
- Calculate answers.
- Classify and hence understand phenomena.

These are important parts of rational thought.

However, we are also faced with problems in this process. The usefulness of standardizing knowledge, so as to make it more authoritative, comes at the cost of whitewashing over these uncertainties. Science itself has to some extent become standardized and even commercialized in curricula today, so that it is often taught misleadingly. Students come to attribute science an almost mystical reverence as they like many in society harbour the erroneous belief that science teaches "truth" rather than approximate models of practical value.

There is a lesson here: shrink-wrapped packaging of education which violates the questioning and critical spirit of scientific discovery can breed ignorance as quickly as it can teach skills. These are important skills in system administration. It is vital that students be shown how to ask fundamental questions and be able to question assumed truths. Nevertheless there is a role for standardization in the language used to describe a subject. A common ontology is as important as shared cultural values in enabling students to communicate without talking at cross-purposes.

Standard Curricula for Computer Science

The Ironman Curriculum effort started by the Association of Computing Machinery is an effort to standardize the framework of topics in a consistent taxonomy. This has been a long term effort. The complete Ironman report includes a number of documents, e.g., [9]. These are far too numerous to discuss here. They can all be found at the ACM website.[1] System administration was initially absent from the Ironman documents, which have been growing since the late 1990s and were finalized only in 2005. Several terms have now entered into the curriculum.

It is interesting to see how terms are integrated into a traditional computer science framework. While some system administration activists would apparently prefer to define system administration as an entirely separate enterprise, here we see that topics have been slotted into the existing taxonomy of categories rather than defining it as a tumour to be tacked on to the edges of computer science. This partly reflects the approach to the subject taken in Oslo, where some compromise on words and terms has been made to integrate the knowledge into the larger picture of programming, Unified Modeling Language design, database searches, service orientation etc. However, there are topics in system administration that do not traditionally appear anywhere else in computer science: fault management, reliability, policy etc. These are also existing disciplines that overlap system administration with other fields, possibly in different university faculties.

On closer examination the course material in our Masters programmes fits (surprisingly) comfortably into the Ironman standard curriculum as long as one reads it with appropriate glasses. This means that even colleges without an explicit degree in system administration could put together a helpful syllabus that would be relevant to the field, with the help of some massaging of language and examples.

From Principles to Content

We turn now to the content of our degree courses. What are the key areas that constitute an education in system administration? Previously the System Administrator's Guild SAGE, with the special assistance of Rob Kolstad, has proposed to build a taxonomy for system administration. Other possibilities for mapping knowledge have been proposed recently with the growth of interest in semantic webs: the concept of ontology is presently quite popular [10]. An ontology goes beyond this with an extended list of terms, usually belonging to a single and uniform cultural body. So far however, this has not formed a useful basis for an educational map where common concepts bind topics together rationally. One reason for this could be that taxonomies are inherently subjective and such subjective impressions and focal points change very quickly in the technological disciplines and create more controversy than consistency.

At Oslo university College, our approach has been to search for a stable core that can be used to teach understandable models and principles, and then to pepper this core with contemporary examples and practice, because understanding is based on models. The course structure was built from the Bachelor level up by identifying common principles from a mass of empirical writing and practice [6]. However, in spite of having courses at Bachelor and Masters level, there are topics that we cannot cover sufficiently for everyone's taste. For example, there is probably sufficient material to give an entire course on "storage" (quite desirable in present times), but this would only mean less time for something more fundamental and the time might simply be wasted when nest year the technologies were different.

We find that, in spite of good intentions, we have a particular difficulty giving students a realistic insight into practice. Some basic skills about practice can be learned through laboratory work and research, other skills must be learned implicitly by reading and writing. We are forced to make a judgment about the best use of a student's time for the long run.

An obvious example where an ontology might help to rationalize our approach is in finding a common set of concepts to be used by UNIX administrators and

[1] http://www.acm.org/education/curricula.html

Windows administrators. Similarly, the chasm of terminology between network management and system administration is so vast that most telecom network management people do not even realise that system administration is a task.

It would be an easy option to give courses in network management, if the idea was simply to fill up a curriculum to attract students. Europe and Asia have dominated the research and commercial activity in Network Management for many years, while the United States has been the greater champion of system administration. Network management has been dominated by the input of software developers using data models to "manage" (usually only to record in a database) information about network devices. The subject is highly protocol oriented and leans towards layered models of centralized management. System administration has been more about UNIX and its highly attractive open environment – giving great freedom to individualists, but consequently lacking focus. An ontological study trying to place knowledge into one cultural framework in such a way that it can be mapped into another, even inexactly, could help to bring these two fields together more quickly.

In putting together courses at our two universities, we have purposely not been led down the paths of least resistance. Rather we have tried to supplement the somewhat bureaucratic views of network management with a more engineering viewpoint. We do not believe that "management" should be equated with monitoring of devices, nor with change management models or database modeling. Instead we have looked for a constructive approach to systems: how to build and maintain them, with a critical eye.

In our chosen course profiles we have effectively proposed our own poor-man's ontologies of most meaningful areas, as many words are inconsistently used, but they are coloured by our own particular tastes and specialties. We shall present some details of these below. As readers will see, the basic ideas chosen by both institutions have emerged along quite similar lines.

Two Year Oslo Programme

In our initial plans for an international two year Masters at Oslo, we expected to be able to require a number of courses in basic system administration related skills, possibly by different names, to the level of our own course System Administration #1. We also hoped also to request some basic computer security. However this wish list soon proved impossible to achieve. With the exception of our own students, there were practically no other student applicants with this kind of background. In security especially, we could see that most colleges offering courses in "security" were really teaching applied encryption, not the kind of rounded security management that we expected. Consequently we were forced to lower our expectations to admit students with any kind of Bachelor in Computer Science, possessing basic university level maths (discrete math, calculus and matrices) and develop a common framework. We altered our priorities to teach students their missing skills during the master programme. Several approaches have since been tried, including intensive catch-up work in the lab; however, we have ended up by offering the missing courses in their entirety as optional modules, since one cannot digest the concepts in a few lab exercises.

The need for a programme that straddles the dual requirements of an academic degree and a strong practical component is undisputed. The scientific tradition in network and system administration is rather weak, and it has been one of the goals of the Oslo research group to strengthen this. As one of the few institutions carrying out scientific research in this field, we are in a unique position to be able to feed the results of current research into teaching. This happens continuously as new developments and technologies emerge.

One possible approach to curriculum development is to take a mercenary approach and give people what they want. When asked what skills graduates should have, employers are quite unclear however. Some employers would like graduates to be ready-trained to begin work installing a Storage Area Network. Others want graduates who can "think for themselves" and see the "big picture." It was left to the college to decide the curriculum.

We recognized that we cannot teach our graduates all of the skills required to be a successful system administrator. What we hope to achieve is for them to think clearly, constructively and critically about problems, to learn for themselves and be independent thinkers. We have consciously avoided naming courses with specific technologies (e.g., LDAP or APACHE) or skills, except perhaps in the case of the Supercomputing course, which is run by a third party.

Skill-training is probably the weakest part of the courses in Oslo. Students are expected to gain practical skills as a by-product of their work in the laboratory. Some but not all students achieve this. The result is that good students learn both practical and analytical skills, while other students tend to excel in only one or the other. On the other hand, we receive clear feedback from students that the subjects that "change their lives" the most are the more scientifically oriented courses, such as the basic laboratory training and Analytical Methods courses, as many of these basic scientific ideas have never been explained to them before as something they could use.

Course degree programmes are naturally hostage to the general trends in world education. These vary from country to country and we see all variations in our international programme. We must cater to quite different attitudes to learning and skill sets.

Goals

A brief summary of course goals at Oslo follows:

1. Graduates should have an insight into the most important technological developments and scientific results in the field.
2. Graduates should have the ability to apply their knowledge and insight in this field.
3. Graduates should be able to think in the abstract about systems, using the idea of models, generalizations and approximations. They should be familiar with traditional terms, philosophies and concepts of modern scientific thinking.
4. Graduates should know how to search the existing literature of the subject.
5. Graduates should have the capacity to communicate clearly both orally and in writing. They should be skilled in giving clear and comprehensible presentations and be capable of explaining their ideas at the appropriate technical level.
6. Graduates should be aware of societal and ethical aspects of the use of computer technology and its management. They should be able to make ethical judgments and argue for these. They should recognize the difference between an opinion argument and a rational scientific judgment.

These goals can be achieved in a number of ways. We divide the qualities amongst broader subject areas.

Research Based Teaching

Our focus on a strong sense of scientific values has been driven by the desire for our research into computer systems to be of the highest scientific standard (a standard in which Oslo university College leads). Our participation in a European Network of Excellence (EMANICS) also feeds directly into our programme and has motivated several modernisations of terminology and minor changes of focus in the topic we cover. The fact that we focus on general principles, illuminated by examples, allows us to rapidly incorporate changes in current technologies and ways of thinking without completely changing the curriculum.

Our course in High Volume Services, on the other hand, was an area in which we responded directly to a need from industry to system administrators with competency in large scale data center deployment. Such a course did not exist anywhere in the world to our knowledge and thus we instigated a number of research projects which formed the basis of this course [11, 12, 13]. See Appendix Course Descriptions from Oslo for courses descriptions.

Text Books

The lack of textbooks was initially a problem, but three books form the core of the principles of the course. The first book newcomers need is Mark Burgess' Principles of Network and System Administration [6]. Without this book, students who have never worked as system administrators have no idea what the subject is about. Later, the Practice of System and Network Administration by Limoncelli and Hogan [14] provides excellent advice about experiential and management matters, but remains difficult for students to understand without experience of working as a system administrator. It is recommended as supporting literature as it is eminently readable. Finally, we use Analytical Network and System Administration along with extensive exercises as the basis for teaching scientific method. This book is too difficult in its presentation however and only excerpts are used in practice. In addition to these core books, we provide students with a library of many technical books on special topics.

Finally, to bring system administration up to the state of the art, we have collaborated in the publication of a new collection of essays by experts in the field of system administration called the Handbook of Network and System Administration [10].

Examination Forms

We require students to read, write, speak and "do" well during their sojourn at the College. The ability to both think and communicate those thoughts is central to our ethos. We test students on their skills in

- Impartial reporting of procedure and results (scientific method).
- Opinion or standpoint formulation (decision-making).

Our preferred examination form is the oral examination combined with course work. This is a very cost-effective approach to gauging students' understanding, as long as student numbers are not too great. By teaching at Masters level, we can make this economically viable to keep a smaller number of students, supported by research activity.

Even students who are initially weak in English language end up being able to make reasonable presentations and have no serious problems in mastering this form of examination, thanks to a consistent emphasis on communication skills throughout the course.

Relationships Between Courses

The course programme was designed with particular care (see Figure 1) to teach concepts, theory and combine this practical experience. Since some concepts require skills that computer scientists are particularly poor in (e.g., calculus, statistics and empiricism) it is especially important to introduce these concepts repeatedly over time. The programme is composed of four semesters which have principal goals as follows:

1. Provide introductory or fundamental concepts, knowledge and skills.
2. To make students independent in their learning, experience self-learning by trial and error etc.
3. To teach students how to view the world analytically using models and experimentation followed by interpretation. Students are encouraged to develop the capacity for original thought.

4. All of the above are implemented in an individual project.

Although Masters degree projects are increasingly being allowed as group efforts in colleges and universities, we strongly encourage students to undertake individual projects. In certain cases projects have been related, but then each student writes an independent thesis.

The figure shows how courses follow on from one another and build knowledge and experience that is reused in later courses. The 'most important' courses are those which collate, integrate or serve knowledge to and from the largest number of other courses. These include the laboratory course and the analytical methods course. Naturally the final thesis itself essentially builds on all of the foregoing courses. However, this depends on the exact nature of the project chosen by the student.

In addition to actual subject matter, there are several cultural delineations of the same material by different industrial and academic subgroups:

- Network Operations community perspective.
- The telecom industry perspective.
- The UNIX community perspective.
- The Microsoft Windows or Apple MacOS perspective.

Although we try to cover all of these viewpoints, with emphasis depending on context, we are notably stronger in some areas than others. In particular our key strengths at OUC are in UNIX. We have found that it is useful to maintain this emphasis in our courses since it makes our curriculum unique, and fills a niche in education that is missing from curricula of other institutions, where Microsoft systems are often covered exclusively.

One Year Amsterdam Programme

The one-year master's degree programme in System and Network Engineering(SNE) is a successful new study programme provided by the Universiteit van Amsterdam (UvA). It is offered in collaboration with the Hogeschool van Amsterdam (HvA). In 2003, an enthusiastic team of experienced system and network administrators with academic backgrounds initiated the process of designing and implementing this degree programme.

Almost immediately, it became apparent that the programme served a previously unmet societal function: the provision of academically trained system and network administrators. In the past years, it has also become obvious that students who have completed the programme successfully have found positions in a wide variety of areas. Nearly without exception, the employers concerned have commented on the valuable knowledge and skills that these students have brought to their work.

This success is partially the result of the programme's design, organisation and philosophy, with special considerations related to the short one-year track.

The following are among the valuable features of the degree programme:

- A strong connection to practice with an approach that is simultaneously theoretical, product-independent and supplier-independent.
- A solid, clearly outlined and well-coordinated programme.
- Strong social cohesion resulting from the large amount of time that students spend together in the SNE Lab.
- Strong motivation and an excellent attitude toward study on the part of the students, partially due to a strict admission procedure.
- Emphasis on concepts (knowledge/insight), with less emphasis on operational procedures.
- A unique design for the final phase of the programme, which is divided into two short (one-month) research projects.

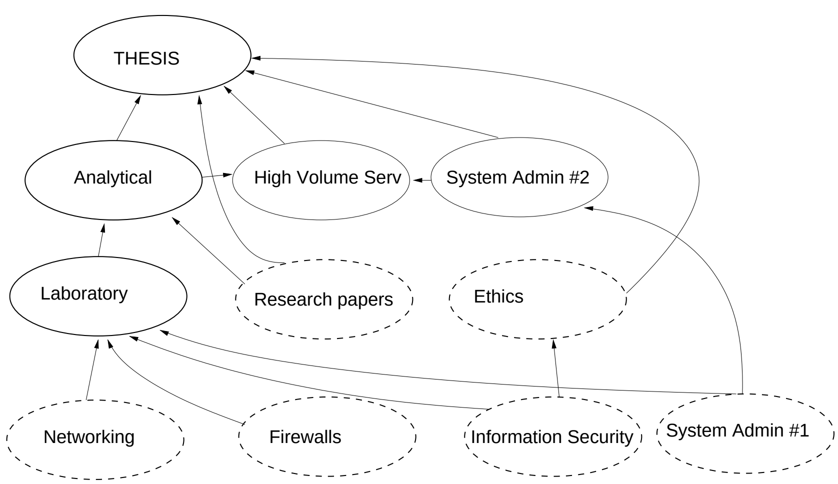

Figure 2: Dotted courses are fundamental sources of information and skills. Heavy lines show the most 'central' courses, i.e., those with the greatest connectivity or impact on collating knowledge. The four levels correspond approximately to the four semesters.

- Considerable emphasis on the preparation of reports and presentations.
- Emphasis on current relevant themes, including open technology and security.
- Strong involvement on the part of (core) lecturers.
- Continuous evaluation and adjustment.
- Strong connection to research, as conducted by the SNE group of the UvA.

At the outset of the programme the name System and Network Administration was chosen. Experiences with all of the classes that have thus far completed the programme and with the employers who have hired the graduates have shown that, in practice, the concept of system administration is not usually associated with academic training. Therefore the name of the programme (but not its contents) has been changed to System and Network Engineering. Engineering is a better pars pro toto, because the programme places considerable emphasis on the architecture of systems and networks, including the engineering aspects, besides administration and management.

For more information about the structure and organisation of the Amsterdam SNE master we refer to the self-assessment made for the accreditation committee visiting in March 2007 [15].

Goals

The Amsterdam education has the same general goals as the Oslo master as detailed in Goals. Some more specific qualifications can be specified as follows:

1. Graduates should be skilled in exploring (searching, reading and evaluating) the many forms of documentation and literature concerning system and network engineering, with regard to both content and medium. They should be familiar with the ISOC, the W3C, IEEE and other international bodies that develop standards and publish in the area of computer systems and networks.
2. Graduates should be very familiar with the usual configurations and procedures for the normal and crisis administration of a variety of current systems and networks, middleware and applications. They should therefore be quickly employable in the usual multi-vendor systems and network contexts.
3. Graduates should be very familiar with the security functions of systems and networks, and they should be capable of contributing actively to the architecture and configuration of systems and networks that conform to current security standards. Graduates should also be able to determine whether systems or networks conform to particular security standards.
4. Graduates should have the technical knowledge of communication protocols, network components and business systems that they will need to accurately justify choices and steps relating to administration and security, including those regarding configuration, procedures and security architecture.
5. Graduates should have sufficient insight into the organisational contexts within which systems and networks function to channel the needs of organisations and users, and to translate them into appropriate technical support.
6. Graduates should have sufficient technical knowledge and intellectual capacity to assume positions of leadership in the field of system and network engineering within a few years. They should have the capacity to develop their own vision of the field of system and network engineering, thus contributing to evolution and innovation in concrete system environments.

Relationship Between Courses

In an effort to realise a coherent programme, two main topics were formulated to serve as a unifying theme for the entire study programme. These topics relate to current and important themes within the profession.

- Open Technology, which involves open standards (e.g., RFCs), open software (including open source) and open security (the antithesis of security through obscurity).
- Security, both technical and non-technical.

With respect to the educational design, the degree programme is based upon a set curriculum, which is the same for all students. The courses are offered according to a set schedule with many contact hours and an attendance requirement. The programme includes four technical-theoretical courses (CIA,[2] SSN, INR and LIA), one non-technical theoretical course (ICP), one 'basics' course (ESA), two practical courses (DIA, IDS) and two research projects (RP1, RP2). The interdependence of these courses is depicted in Figure 3.

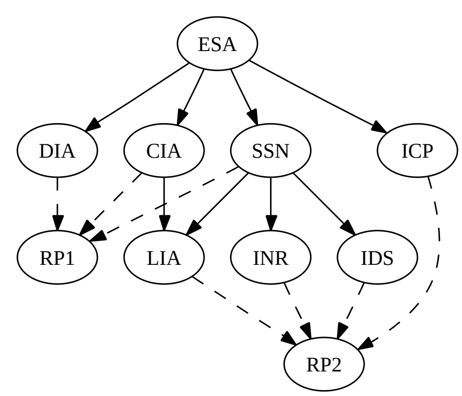

Figure 3: Interdependence of SNE courses.

This approach, according to which students follow largely the same programme, strengthens the coherence of the programme and the commitment of the

[2] For an explanation of the course abbreviations, see appendix Course Descriptions from Amsterdam.

students to the study programme. It encourages students to attend and participate in the study components, and it has a positive effect on academic achievement.

The programme is continuously evaluated through discussions with students and through regular meetings by the lecturers, thereby improving the coherence of the various components.

The curriculum involves a gradually increasing level of difficulty and during the course of the year students work more and more independently. This effect can be clearly observed in the research projects, within which the quality of the work that is submitted at the end of the year is clearly superior to that of similar work that was submitted halfway through the programme.

Programme Comparison

It is instructive to compare the approaches in the two programmes carried out for the university Accreditation committee for the master education SNE at the Universiteit van Amsterdam. We reproduce their table here, see Table 1, showing the correspondence. It is interesting to see that both courses emphasize the same basic ideas, even though the mode of implementation is somewhat different.

Course Implementations

Oslo

The implementation of any new programme presents a number of challenges both in terms of resources and imagination. We have now seen the progress of four groups of students at our college and are able to draw a number of experiences from this time. The opportunity to craft the entire curriculum without interference from national academic title regulation has also made the course curriculum highly cohesive and integrated in comparison to the often poorly cohesive topics at Bachelor level. Students notice this difference and have commented on it on several occasions in evaluation meetings.

The strong international nature of our group, both in the staff and the composition of the students, makes an attractive milieu for our students. We have used teaching staff from the UK, Norway, Somalia, USA, and Switzerland including our fixed members. Our students come from all over the world, including Asia and the Far East, Africa, America, and Europe. Our English speaking environment works for the most part well, and the support from the admissions office has been excellent. Our active programme of Internationalization has brought visitors from America, The Netherlands, and a number of countries participating in our European Union Network of Excellence EMANICS.

Our college is especially strong in the area of UNIX and GNU/Linux technology. This gives our students a clear advantage in service management as many server room installations are still based on UNIX technology. We have chosen to maintain this focus rather than dilute it with 'more common' Microsoft technology because it gives the students an advantage. They can learn the Microsoft technologies themselves based on their UNIX knowledge We aim rather to show how to integrate these different technologies.

A weakness of our course initially was that it did not cater well to students who lacked the assumed prerequisite experience. Such ideal students were near impossible to recruit however, and so we have adapted the initial part of the course to arrange for essential background skills to be taught. The students we typically attract:

- Are looking for a way to work with computers that is not about programming.
- Have rarely any background in UNIX.
- Increasingly have not worked as system administrators either, but are attracted by the term

Oslo	ECTS	Amsterdam	ECTS
Computer and Information Security	10	Security of Systems and Networks	6
Networking: Technologies and Principles	10	InterNetworking and Routing	6
Intrusion Detection and Firewalls	10	Intrusion Detection Systems	6
Network Infrastructure and Security Laboratory	15	Assignments	
Research papers	5	Assignments	
Optional subjects	10	Distributed Internet Applications	6
		Essential Skills for Administrators	6
Analytical System Administration	10	Large Installation Administration	6
Network and System Administration #2	10	Classical Internet Applications	6
Social Aspects of Systems and Ethics	10	ICT and Company Practice	6
Final Thesis	30	Research Projects 1 and 2	6/6
	120		60

Table 1: Comparison of emphases between Oslo and Amsterdam masters courses. ECTS are European standard study points, denoting course length.

"Network" as made popular by the spread of the Internet and broadband.

One of the major strengths of our programme is a strong course in applied scientific modeling, not merely philosophy of science, but analysis, understanding and application of mathematical representations. All students are sceptical of the analytical, mathematical and scientific content we expect in the course, but many if not most students later claim that it ends up as being one of the most valuable courses to them, teaching them how to think about problems in a way that they had never learned before during their university careers.

In a recent yearly meeting of the steering council, a student representative commented that he was learning more in our programme than he ever had before. When asked why, he replied: "Because we have to." The students regard our courses as demanding, and they see this as a positive quality.

A weakness we have identified in our course is the lack of realistic experience in the management of computer systems by the students. This is something that we have striven to simulate but without much success. Limitations in laboratory space and user activity have proven to be insurmountable problems for us. One way around this problem is to use *internships* where students spend their summer working in a local company. The practice of internships has not been widespread in Norway however, as the Norwegian summer holiday culture usually means that almost no one is left in companies in the summer months to provide guidance for students, and hence none are taken on. However, most recently we have made some progress in this area too as Norwegian society adapts.

Amsterdam

As mentioned in section on the relationship between courses, the setup of the master programme is a coherent effort centered around the themes of Open Technology and Security. "System" topics and "Network" topics are considered to be of equal importance and are treated as a unified duality throughout the year. The program is completely fixed – apart from the obvious choices in subject for the research projects – and courses are not accessible for students outside the programme. This didactic concept creates a very tight social relationship between all students and is the basis for a kind of melting pot in which teachers and students together get the maximum possible result from the efforts put into the programme. It is the firm belief of the educators in Amsterdam that such a concept is indeed necessary to be able to graduate within only one year at a sufficiently academic level, entering from a higher professional education level. The didactic concept alone is not sufficient to succeed. Students also need to belong to the top layer of the higher professional education and bring along an excellent motivation to be able to cope with the demanding year, both in time as in energy to be invested.

A major contribution to the melting pot is the so-called SNE Lab. This is a room suited for multiple purposes, accommodating about 20 students with a lecture room for centralised teaching, a desktop environment for each student, a dedicated network for production and experiments, a dedicated VM-based server for each student for projects and experiments and some space for social activities. A major part of the total study time is spent in the SNE Lab, also contributing to better graduation rates and opportunities to help each other out in case of problems.

As is the case in Oslo, students do not always have the prerequisite experience. Unconditional acceptance into the master's degree programme in SNE is possible for students that have successfully completed one of the university bachelor's degree programmes in Information Science. Admission into our vocationally oriented master's degree programme is also possible for students that have successfully completed a programme at an institution of higher professional education (HBO), on condition that they pass an extensive intake examination or assessment. In practice, the majority of applications appear to come from students who have completed HBO degree programmes. The following components are part of the intake procedure:

General Skills

- Literature skills: reading and summarising a technical document.
- Oral skills: presenting a previously written thesis.
- Analytical skills: basic knowledge of discrete mathematics.

Specific Skills

- Basic knowledge of UNIX.
- Basic network knowledge (TCP/IP).
- Basic knowledge of scripting (shell).

Within the actual programme, the ESA course offers students the opportunity to brush up on the knowledge and skills that they will need to be able to follow the rest of the programme successfully.

The master does not result in a regular thesis project. The one-year format demands another construction here. Most students have already written a thesis for their bachelor's. The competence added for the SNE master is the ability to research an academic topic in a very short time (one month) and write a consultancy report about it. All learned skills and knowledge come together within these projects as far as researching, communicating, writing and presenting is concerned. A disadvantage of this setup is that the scope of research has to be restricted to fit within the one month period. An advantage is that producing timely and short reports giving insight into a problem is of great practical advantage in real life situations. Moreover this format creates room to exercise these skills twice a year (in January and in June).

An often heard remark made by students is that they learn in a short time more in our education than in years before in other educations. This is almost certainly due to the didactic concept and also shows that students are able to perform better under the right circumstances.

Our education is in Dutch, because we do not aim at recruiting international students right now. A larger group than 20 students would also put stress on the didactic concept. All teaching material however is in English and a thorough understanding of English in listening, reading and writing is necessary. Most study material is presented in the form of slides, accompanying the lectures, or is available online on the Internet. For some courses we make use of books of renowned authors in the subject field like [16, 17, 18].

Employment of the Students

Oslo

Our former students have worked in a variety of locations. A few of these are listed below.

- Norwegian Cancer Research Fund
- Opera Software
- IBM Norway
- The Armed Forces Computer Facility
- Oslo university Computer Operations Center
- Telia
- Norsk Hydro
- Oslo Data Center (ODS)
- Our group (as system administrator and teacher)
- Freecode
- Linpro AS

Amsterdam

Our alumni have a very good prospect as much wanted professionals in a variety of employment opportunities. To name a few:

- Continued research in a Ph.D. position
- Consultancy in small and large advising firms
- Lecturer in higher education
- Network Operations with ISPs and big infrastructure providers
- ICT management in small and large companies
- System architect and engineer
- Security specialist and auditor

Summary

This paper is a statement of impressions. It is clearly too early to make any scientific evaluation of the success of our experiments. Nonetheless we are confident enough in our results to report on our activities and encourage others to follow suit. The empirical quality of our reporting will improve as more years of experience can be collated. With four year of experience we can only assert that, subject to minor adjustments, our programmes have succeeded well in their general goals, but that there is still room for improvement in the details.

We struggle occasionally with resources, but we have identified a strategy for rationalizing teaching burdens by expanding the curriculum marginally.

We continue to monitor and discuss the results of our programmes between our universities, making adjustments on the fly. We believe that these programmes have been a success not only in terms of numbers of students emerging from the conveyer belt, but also in generating an academic cohesion within our groups, stimulating research, and attracting doctoral students, post-docs and international visitors who have enriched our computer science milieux for everyone.

Acknowledgement

We would like to thank the other staff members at our institutions who are involved in teaching and developing the materials for these courses. Naturally we also thank the many talented students who have passed through our corridors and made the teaching worth-while). Mark would like to acknowledge Steve VanDevender for interesting discussions about system administration education that helped to motivate this paper. MB is supported by the EC IST-EMANICS Network of Excellence (#26854)

Appendix: Course Descriptions from Oslo

Network and System Administration I

The aim of this course is to provide an understanding of the role and procedures carried out by network and system administrators. It outlines general principles, while providing concrete hands-on examples using [6].

Computer and Information Security

An introduction to the theory and concepts of security, as applied to computer systems. The course builds on the earlier course on system administration, where security was discussed from a practical viewpoint. A deeper understanding of the principles and examples of security is explored, going beyond the encryption style of many courses on security. Common problems in software design are also noted.

Networking: Technologies and Principles

Explaining the principles and practice of networking technologies and protocols that are used today, for transferring and routing information between hosts. Analysis is a key part of understanding networking, and students are expected to develop a sufficient understanding of the physics and mathematics of communication and traffic flow to gain full marks.

The course introduces a theoretical foundation for the coming laboratory course, and the exercises teach some practical and diagnostic skills in using these protocols, including tools such as traceroute, and the router operating systems IOS/JunOS.

Intrusion Detection and Firewall Security

This course introduces the fundamentals of TCP/IP network monitoring at the packet level and its application to computer security. Detecting and protecting against hostile network activity has become one of the important topics of Network and System Administration.

Social and Ethical Aspects of Systems and Research

Originally two separate courses, this module now consists of two courses under one umbrella. In both these modules students are expected to read and write about system administration. In research papers, they read the academic literature published in the field and summarize it. This is an important skill for the master thesis and for comprehension in a professional setting. In the Social and Ethical component, students are expected to formulate a standpoint of their own and argue for it. This develops communication skills, comprehension skills and powers of rational judgment.

Network Infrastructure and Security Lab

The aim of this course is to give students substantial experience in using networking and system administration equipment in as realistic an environment as we can create. Students should become proficient at handling hardware and software, as well as learn debugging skills and scientific methodology. Experiments carried out in the lab must be documented as scientific experiments and written up in clear, concise English. Presentation skills contribute to the grade, as well as documentation of analytical ability and systematic work practice. Students are graded on performance, reporting, tidiness and safety.

Network and System Administration II

This is a course in essential services and software tools and application services such as DNS, NFS, SMB etc. It includes understanding integration of Microsoft and Macintosh OS and network models, evaluating and gaining experience of data backup and archiving models. Directory Services, Database Administration, automation of maintenance of a LAN with Cfengine are covered. Economical aspects of system management in planning and resource management are discussed.

Analytical Methods for Systems

The aim of this course is to provide a deeper perspective on systems and their administration from a theoretical and cultural viewpoint. Only such a background can stimulate research and creative solutions in the future. The course places system administration in the contexts of other subjects. Students must master some basic modeling techniques and be able to apply simple mathematical methods to problems of planning and analysis of human-computer systems. The results are applied to business process modeling also.

High Volume Computing Services

The aim of this course is to cover the fundamentals of implementing scalable computing systems for parallel computation and service delivery. This includes an understanding of the science of data centers where large installations are kept, and understanding the meaning of scalability. Students should know the difference between High Performance Computing, High Availability Computing and High Volume Computing and know how Amdahl's law applies to parallelization "speedup" and load balancing. Some basic queueing theory is learned.

Supercomputers and Virtual Operating Systems

This course is designed to fill a specific need for the computing industry: experience and understanding of supercomputers and virtual operating systems. Specific attention is given the IBM operating systems zOS (OS/390) and VM that are widely deployed in finance and banking sectors but which are rarely covered in university curricula.

Final Thesis

A one semester project, planned and executed by the student.

Course Descriptions from Amsterdam

Essential Skills for Administrators (ESA)

This course forms the foundation for much of the daily work of a system administrator. If the use of open standards and open source software is to be advocated credibly, system administrators must adhere to these standards as well. In the area of documentation, attention is paid to (pdf)(La)TeX, and XHTML is considered for Web purposes. Version tools, including RCS, CVS and SVN are also addressed, as is the use of secure remote log-in (SSH) and secure communication (PGP, GPG). Finally, a number of scripting languages (shell, Perl, Python, Tcl/Tk and Ruby) are discussed. Written reporting is an important component of the course.

Classical Internet Applications (CIA)

The aim of the course is to understand basic architectural issues in classical client-server environments. Topics covered are:

- Historical awareness of the development of Internet and UNIX.
- Insight into and knowledge of the most important classical client-server applications (DNS, Email, Web and Directory Services).
- Understanding the role of security in designing systems that must carry out the identified services.

Security of Systems and Networks (SSN)

Systems are secured according to a variety of principles, including plain-text passwords, one-time passwords, encrypted passwords, public/private keys and certificates. Networks are secured with firewalls and encryption on the network layer. The topics that are addressed in this course include remote access using SSH, secure Web transactions using SSL/TLS, single sign-on using Kerberos, secure email using PGP/GPG, IPsec and key management. The course also considers the problems of wireless access and WEP. Many of the security systems that are mentioned for these purposes are based on the encoding of information (encryption). The course also addresses the (mathematical) principles of cryptography. The following skills receive special emphasis:

- Evaluation of security technology.
- Cooperation in groups of two and four.
- Independent literature searches.
- Written reporting.

Distributed Internet Applications (DIA)

The issues concerning the development of middleware systems for large-scale computer networks are discussed. Principles taught include communication, processes, naming, consistency and replication, fault tolerance, and security. These principles are further explained by means of different paradigms applied to distributed systems: object-based systems, distributed file systems (NFS), document-based systems (the Web), and coordination-based systems (publish/subscribe systems). Explicit attention is paid to the practical feasibility and scalability of various solutions. For this reason, experimental (research) systems as well as commercially available systems are discussed.

InterNetworking and Routing (INR)

This course is all about the world of ISPs and transport providers in the Internet. Topics discussed are:

- Mathematical modeling of addressing and routing in the Internet (both IPv4 and IPv6).
- Insight into and knowledge of the abstract and concrete algorithms that are used in routing systems.
- Insight into and knowledge of the virtualisation techniques that can be used to study networks and their routing systems (focusing on RIP, OSPF and BGP).

Large Installation Administration (LIA)

The daily tasks of system administrators and the concepts with which they should be familiar are the focus of this course, which addresses the design, implementation and documentation of procedures for daily administration. Security, stability and manageability are primary requirements in this regard. The course addresses account management, storage management and version management. Particular attention will be paid to the administration of complex systems and networks in large organisations. The course also addresses ITIL, PRINCE2 and other technologies. Finally, presentation skills are emphasised in this course as well.

Intrusion Detection Systems (IDS)

This course focuses on methods and techniques to detect anomalous behaviour, to report on it and to take appropriate measures. Topics included are:

- Examination of hacker techniques and the study of security systems from a hacker's perspective.
- Detection techniques, including Intrusion Detection Systems and Honeynets.
- Protocol analysis and knowledge of tools.
- Rootkit and malware technology.
- Cooperation in groups of three or four and submission of a proof of concept.
- Independent literature searches.
- Ethical aspects of security and network technology.
- Presentation of research results and written reporting.

ICT and Company Practice (ICP)

Master-level positions within organisations require the capacity to form and defend well-founded opinions concerning business-related ICT issues. Those who hold such positions serve as discussion partners for a wide range of managers, advisors and policy makers, many of whom are not technically oriented.

A well-founded opinion is formed through the acquisition of knowledge. The ability to defend an opinion requires good communication (by means of presentations and written reports).

The objectives of the ICP course can be formulated as follows:

- The acquisition of knowledge concerning business-oriented ICT issues. Relevant topics are sourcing, legacy and information security. These topics are chosen specifically because of their interface with the work and responsibilities of system and network administrators.
- Critical evaluation of a non-technical scientific article in the area of ICT.
- Writing a non-technical scientific article in the area of ICT.

Research Projects 1 and 2 (RP1, RP2)

The course objective is to ensure that students become acquainted with problems from the field of practice through two short projects, which require the development of non-trivial methods, concepts and solutions. Each course is a very intensive one-month full-time activity. After these courses, students should be able to:

- Transform a roughly outlined problem into a carefully defined question, supported by literature on the topic.
- Establish a feasible project schedule for answering the question.
- Conduct autonomous research to answer the question at hand, using literature searches, studying, experimentation and/or the development of software and hardware.
- Present solutions to a diverse audience (experts as well as non-experts).
- Defend solutions in debates.
- Provide an appropriate report that is useful to a client.

Author Biographies

Mark Burgess is professor of Network and System Administration at Oslo University College. He was the first professor with this title. Mark obtained a Ph.D. in Theoretical Physics in Newcastle, for which he received the Runcorn Prize. His current research interests include the behaviour of computers as dynamic systems and applying ideas from physics to describe computer behaviour. Mark is the author of the popular configuration management software package cfengine. He made important contributions to the theory of the field of automation and policy based management, including the idea of operator convergence and promise theory. He is the author of numerous books and papers on Network and System Administration and has won several prizes for his work. Reach him electronically at Mark.Burgess@iu.hio.no .

Karst Koymans holds an M.Sc. in Mathematics from Utrecht University and a Ph.D. in Mathematical Logic also from Utrecht University. His thesis is titled "Models of the Lambda Calculus." Karst Koymans has been working as an Assistant and Associate Professor at Utrecht University and the University of Amsterdam. During that period he gained extensive experience with System and Network Engineering, mostly by being network and system administrator for his faculty. Since 2003 he is the director of the master education System and Network Engineering at the University of Amsterdam.

Bibliography

[1] Herzog, U., "Network Planning and Performance Engineering," *Network and Distributed Systems Management (Edited by M. Sloman)*, p. 349, 1994.

[2] IFIP/IEEE Conference, *Network Operations and Management Symposium*.

[3] IFIP/IEEE Conference, *Integrated Network Management*.

[4] IARIA Conferences, (various), http://www.iaria.org .

[5] IFIP/IEEE Conference, *Distributed Systems Operations and Management*.

[6] Burgess, M., *Principles of Network and System Administration*, J. Wiley & Sons, Chichester, 2000.

[7] Burgess, M., *Analytical Network and System Administration – Managing Human-Computer Systems*, J. Wiley & Sons, Chichester, 2004.

[8] Hasle, Thor E., *Nettverksadministrasjon, 2nd Edition*, Cappelen Akademisk Forlag, Oslo, 2003.

[9] ACM College Education Committee, *Guidelines for Associate-Degree Programs to Support Computing in a Networked Environment*, Technical Report, 2000.

[10] Bergstra, J. and M. Burgess, editors, *The Handbook of Network and System Administration*, Elsevier, 2007.

[11] Burgess, M. and G. Undheim, "Predictable Scaling Behaviour in the Data Center with Multiple Application Servers," *Lecture Notes on Computer Science, Proceedings 17th IFIP/IEEE Distributed Systems: Operations and Management (DSOM 2006)*, Vol. 4269, pp. 9-60, Springer, 2006.

[12] Burgess, M. and S. I. Ulland, "Uncertainty in Global Application Services with Load Sharing," *Lecture Notes on Computer Science, Proc. 17th IFIP/IEEE Distributed Systems: Operations and Management (DSOM 2006)*, Vol. 4269, pp. 37-48, Springer, 2006.

[13] Björnstad, J. H. and M. Burgess, "On the Reliability of Service Level Estimators in the Data Center," *Proceedings 17th IFIP/IEEE Integrated Management*, submitted, Springer, 2006.

[14] Limoncelli, T. and C. Hogan, *The Practice of System and Network Administration*, Addison Wesley, 2003.

[15] Koymans, C. P. J., et al., "Self-Assessment of the Degree Course System and Network Administration," Educational Institute Information Sciences, Universiteit van Amsterdam, https://www.os3.nl/_media/2006-2007/accreditatie.pdf .

[16] Speciner, Mike, Charlie Kaufman, Radia Perlman, *Network Security: Private Communication in a Public World, 2/E*, Prentice Hall, 2003.

[17] Steen, Maarten van, Andrew S. Tanenbaum, *Distributed Systems: Principles and Paradigms, 2/E*, Prentice Hall, 2007.

[18] Perlman, Radia, *Interconnections: Bridges, Routers, Switches, and Internetworking Protocols, 2nd Edition*, Addison-Wesley Professional, 2000.