Computer Science
at the University of Helsinki 1998

Greger Lindén and Martti Tienari (eds.)
Contact information

Postal address:
   Department of Computer Science
   P.O. Box 26 (Teollisuuskatu 23)
   FIN-00014 University of Helsinki
   Finland

Email address: postmaster@cs.Helsinki.FI (Internet)

URL: http://www.cs.Helsinki.FI/

Telephone: +358 9 708 51

Telefax: +358 9 708 44441

ISSN 1238-8645
Helsinki 1998
Helsinki University Printing House
Abstract

The Department of Computer Science at the University of Helsinki is the largest such department in Finland. In this report, we present the activities at the department. We describe ongoing research and research projects as well as give a list of recent publications of the faculty according to their classification. We also present the faculty members and note their research interests as well as publications and contact information. We describe the educational program, graduate schools and courses taught at the department and also give a list of accepted theses and abstracts of the most recent Ph.D. theses. We give short descriptions of the department library and the computing facilities, and finally present an overview of our international relations.

Computing Reviews (1991) Categories and Subject Descriptors:
A.1 Introductory and Survey
A.2 Reference
K.3.2 Computers and Education: Computer and Information Science Education

General Terms:

Additional Key Words and Phrases:
computer science education, research information
# Contents

1 Overview  1
2 Research  5
  2.1 Review  5
  2.2 Projects  7
  2.3 Publications  22
3 Faculty  57
4 Education  75
  4.1 Educational Program  75
  4.2 Graduate Schools  78
  4.3 Course Descriptions  82
  4.4 Accepted Theses  91
  4.5 Abstracts of Recent Ph.D. Theses  99
5 Library  105
6 Computing Facilities  107
7 International Relations  109
Chapter 1

Overview

The Department of Computer Science was founded in 1967 when the first full professorship in computer science was established at the University of Helsinki. It is within the Faculty of Science, along with the departments of Mathematics, Physics, Chemistry, and others.

Our current teaching faculty (May 1998) comprises 46 full time teachers. It can be categorized using American academic terminology: 5 full professors (Mannila, Tienari, Ukkonen, two vacant positions), 4 associate professors (Mäkelä, Paakki, Raatikainen, Sippu), 23 assistant professors (9 research oriented senior assistants and 14 teaching oriented lecturers) and 14 teaching assistants. We also employ approximately 40 of our students in teaching on a part-time basis. Fourteen senior experts are also associated with the department. These so-called docents work mainly outside the university but occasionally give courses or supervise theses in the area of their speciality. Approximately 60 research positions are financed from outside sources. We also have a staff of 12 persons.

The department annually admits 240 students to major in computer science. The students are selected according to their standing in a national student examination or in a special entrance examination (or both). The number of completed M.Sc. degrees (5 year degree) was 44 in 1995, 51 in 1996, and 62 in 1997. The study time for a M.Sc. degree ranges from five to eight years. Many of our students work in industry, which slows down or stops the progress of their studies. Fairly frequently our students, after having acquired the basic skills in computer science, redirect their studies by transferring to the Helsinki University of Technology, the Helsinki School of Economics, Faculty of Social Sciences, or some other educational institution. Many students study computer science as a minor while pursuing a major in another subject, such as mathematics, physics, economics, psychology, or social sciences. We offer two curricula for students minor-
ing in computer science. In 1997 our “approbatur curriculum in computer science” (15–34 credit units) was completed by 141 students and our “cum laude curriculum in computer science” (35–69 credit units) was completed by 117 students.

There are two graduate degrees in Finland: the Ph.Lic. (3 years) and the Ph.D. degree (4 years). The latter has higher quality requirements. Both build upon the M.Sc. degree (5 years). The high demand for our M.Sc. graduates to fill well-paid jobs in industry is a fact which has hampered our Ph.D. education. Our department granted five Ph.D. degrees and five Ph.Lic. degrees in the two year period 1996–97. In postgraduate education we cooperate with the Helsinki University of Technology within the Helsinki Graduate School in Computer Science and Engineering (HeCSE) and with the University of Turku and the Center for Scientific Computing of Finland within the Graduate School in Computational Biology, Bioinformatics, and Biometry (ComBi).

Three principal sources provide funds for computer science research in Finland. The Academy of Finland under the Ministry of Education and Science provides funding for basic research. The second important research financier is the Technology Development Centre (TEKES) under the Ministry of Trade and Industry. We also enjoy financing of the European Commission in the research and development programs ESPRIT and ACTS.

The department maintains jointly with the University Computing Centre a good computer science library. It subscribes to most major international journals in computer science and related fields and acquires a majority of the most important computer science books and conference publications. The library is run by a librarian and a secretary.

The University Computing Centre—http://www.helsinki.fi/atkk/ maintains a communication backbone network and offers UNIX and PC services. In addition, the department maintains its own workstation network of approximately 280 Linux PCs and about 10 servers. Windows 3.1, Windows 95 or Windows NT can be used as an alternative for Linux. About 30 of the Linux workstations are mobile laptops which can join and leave the network dynamically. Classrooms with Linux PCs are available. Each office of the department has a PC or a workstation.

The department has three informal sections that are used in the planning of the curricula and in administration. The division is not strict, and several research projects span two sections. The sections cover roughly the following subject areas:
1. General Computer Science (Prof. Esko Ukkonen, Assoc. Prof. Matti Mäkelä): algorithms and data structures, computational complexity, computational geometry, machine learning, Bayesian networks, neural networks, computer graphics, numerical and symbolic computation, computational biology, geoinformatics, computationally intensive tasks, computer-aided instruction, computers in education

2. Computer Software (Prof. Martti Tienari, Assoc. Prof. Jukka Paakki, Assoc. Prof. Kimmo Raatikainen): programming languages, compilers, formal specification and verification, software engineering, distributed systems, computer networks, operating systems, performance evaluation

3. Information Systems (Prof. Heikki Mannila, Assoc. Prof. Seppo Sippu): databases, human-computer interfaces, computer supported co-operative work, information system design methodology, design of databases, text databases, object-oriented databases, logic databases, database structures and algorithms, document management, data mining and knowledge discovery, management of spatial data (GIS).

The University of Helsinki has many diverse teaching and research offerings related to computer applications. At the Department of Mathematics there is an active group in mathematical logic, numerical analysis and symbolic computation as well as some interest in theoretical computer science. The Rolf Nevanlinna Institute under the Faculty of Science is a research Institute of mathematics, computer science, and statistics with the main tasks of research and doctoral education. Our students also benefit from the hardware-oriented teaching (e.g. electronics, digital electronics, microcomputers, interface electronics) given at the Department of Physics. In the Faculty of Social Sciences some teaching and research is devoted to computational statistics, administrative information systems, and the social effects of data processing. In the Faculty of Arts there is a research unit in computational linguistics and a degree program in linguistic theory and cognitive science.
Chapter 2

Research

2.1 Review

The research at the department has evolved over the years similarly to the international research trends in computer science. Early work in numerical analysis in the 1960’s made room for work in programming languages and compilers in the 1970’s. Since then the research has diversified and its volume has increased. In the following, the research activities of each section of the department are reviewed.

The main research areas in the section of General Computer Science are algorithms and data structures, machine learning, probabilistic reasoning, computations by complex dynamic systems (cellular automata and genetic algorithms) and computational biology. Algorithms and data structures is the area with the longest tradition. The work on string matching algorithms (Ukkonen, Tarhio) has been particularly successful. Theoretical work has often been conducted within the framework of systems research providing practical motivation for the problems studied. Currently, special emphasis is given to the research on algorithmic problems in computational biology and bioinformatics.

Machine learning and probabilistic modeling are active research directions related to artificial intelligence. The complex systems computation group (CoSCo) has studied prediction and model selection issues in the probabilistic and information-theoretic frameworks, focusing on probabilistic model families, such as Bayesian networks and finite mixture models. The group has also performed extensive empirical and theoretical work on stochastic optimization methods, including simulated annealing and genetic algorithms. The machine learning group (Kivinen, Manila, Ukkonen) has studied different learning models and the complexity of learning tasks within these models. One of the results is the first
MDL learning algorithm that has a proven performance guarantee. The more practice-oriented work has developed, for example, new Occam algorithms for learning decision trees and decision lists, and software tools for testing and comparing various machine learning algorithms.

Our computer software research can be subdivided into two main areas: distributed systems and telecommunication software (Raatikainen, Tienari) and programming languages and software engineering (Paakki). In some projects these two areas are intertwined.

In telecommunication software we have both industrially and academically oriented research projects. In the former category one could classify the European ACTS-project DOLMEN (Service Machine Development for an Open Long-Term Mobile and Fixed Network Environment, Raatikainen) as well as the project Mowgli (Tienari, Raatikainen, Alanko) concentrating on mobile computing in general and also in developing a software architecture for implementing mobile-aware applications. Also the project RODAIN (Real-Time Object Based Database Architecture for Intelligent Networks, Raatikainen) has strong ties to the Finnish telecommunication and software industries. A new project on agents in mobile computing has just started.

The projects in distributed systems are OCDE (Tienari), investigating open distributed computing environments, and MOCO (Modeling Concurrency, Tienari), concentrating on formal specification and verification of distributed systems. In the latter area, in particular, we have achieved notable results, well-received by the international research community.

The department has long traditions in the research of programming languages and compilers. This research still continues but is mainly directed now to object-oriented languages and their implementation problems. From this work a new research line in software engineering (Paakki) has evolved. There are two projects in this area, A Channel into Object-Oriented Protocol Design and A Framework Editor for Java (FR ED).

In information systems the largest research project concentrates on data mining (Mannila, Toivonen, Verkamo), also known as knowledge discovery in databases. The work on data mining has several subprojects including industrial projects as well as an ESPRIT project. The research is done in cooperation with the machine learning group, with statisticians, and with the appliers of the work. Recent research results include efficient data mining methods for database re-engineering, methods for finding recurrent episodes within event sequences, and development of automatic tools for the simulation of complex statistical models.

Other research projects include document management (Kilpeläinen,
2.2 Projects

Mannila), computer-supported cooperative work (Erkiö), and animation of algorithms (Tarhio). Research results include new efficiently evaluable query languages for text databases and methods for assembling structured documents.

Active projects and research areas as well as individual research of some researchers and graduate students are presented in more detail in Section 2.2. Section 2.3 lists a selection of recent publications. More information on research projects can be found on the WWW pages of the department.

2.2 Projects

a) General Computer Science

Algorithms on Strings

Research on strings algorithms started at the department quite early, in 1981. The initial impulse came from computer applications in molecular genetics where there is a lot of demand for efficient and sophisticated string-manipulation algorithms. The group is now one of the leaders in its special field in the world and has obtained several basic results that have been included in recent international text books.

The reputation of the group is based on the work done in the 1980’s on developing new algorithms for the following problems: edit distance, approximate string matching, DNA sequence assembly, and shortest common superstring. In the 1990’s, various sublinear approximate string-matching algorithms and new measures for string similarity have been developed. For the important problem of how a text should be preprocessed to speed-up subsequent approximate string searches, several solutions have been discovered based on the suffix-tree of the text or the so-called q-grams. For constructing suffix-trees, a natural on-line algorithm has been developed. The two-dimensional string matching problem has also been studied. We were able to obtain the first optimal expected time solutions using simple, practical algorithms.

The current research topics include feature-based algorithms for approximate string matching, two and higher dimensional string matching with applications in computational biology (modeling of viruses, protein folding), sequence databases with applications in bioinformatics, the indexing and clustering of strings and documents, and the search by content in image and music databases.

Current members of the group are Prof. Esko Ukkonen (group leader),
Machine Learning

Being able to build computer systems that can learn in some sense is one of the very central problems in artificial intelligence. Inspired by certain theoretical advances in the field such as Valiant’s PAC model and Rissanen’s MDL principle, the project generally aims to apply to machine learning the approach of theoretical computer science and algorithmics. This international trend is called computational learning theory. Our theoretical work is supported by experimenting with the algorithms.

Results on decision tree learning include analyses of the requirements for efficient multisplitting on numerical attributes and comparison of the most commonly used attribute evaluation functions with respect to these requirements. A practical method that results in optimal splitting has also been developed. Testing environments for learning algorithms have been developed.

Some promising results have been obtained about MDL based learning algorithms in particular for the case where the instances are strings of arbitrary length. The method has been applied to a clustering problem of biological sequences.

Research has also been done on on-line learning algorithms that make no assumptions about the distribution of noise in the data. This is known as agnostic on-line learning. Emphasis has been on learning simple statistical models, such as generalized linear models, with algorithms that learn fast even if there is a large number of irrelevant variables present.

The current research themes of the group are:

- development of MDL-based and other learning algorithms with a particular emphasis on potential applications in computational biology,
- development of decision tree learning algorithms that have a sound theoretical basis and perform well in practice,
2.2 Projects

- generalizing the agnostic on-line learning methods further and deepening the understanding of their relationship with more classical statistical approaches,

- applications of symbolic learning approaches to problems of mobile robotics, and

- (lazy) learning for planning in continuous domains, with applications to an industrial case-based bioprocess planner (joint work with VTT and Finnish breweries).

The group has good international reputation and is together with nine other European groups a member of the ESPRIT Working Group Neuro-COLT II and a site of European Machine Learning Network. The group also works in close co-operation with Prof. Heikki Mannila’s data mining group.

The members of the group are Prof. Esko Ukkonen (group leader), Doc. Jyrki Kivinen, Dr. Tapio Elomaa, M.Sc. Tibor Hegedüs, M.Sc. Markus Huttunen, M.Sc. Juho Rousu (VTT Biotechnology and Food Research) and M.Sc. Jaak Vilo. The group gets funding from the Academy of Finland and from the European ESPRIT Programme.


Probabilistic Modeling and Complex Systems Computation (CoSCo)

A complex system is a collection of simple interacting agents, elements or processes, whose collective behavior exhibits interesting large scale phenomena. Such systems can be found in various disciplines, including computer science, economics, mathematical biology and physics. The Complex Systems Computation (CoSCo) research group studies computational issues related to complex systems focusing on prediction and model selection issues. Current work of the CoSCo group is concentrated on theory and applications of Bayesian (belief) networks, and related probabilistic model families, such as finite mixture models. Other research areas addressed include case-based reasoning, artificial neural networks, and stochastic optimization methods, such as simulated annealing and genetic algorithms. The results achieved include extensive theoretical and empirical studies concerning

- efficient methods for learning probabilistic models from sample data;
• a novel, computationally efficient Bayesian criterion for selecting the most relevant model features (attributes);

• the accuracy of different marginal likelihood approximation techniques for Bayesian networks with hidden variables;

• similarities and differences between Bayesian and information-theoretic (MDL, MML) modeling approaches;

• a general Bayesian framework for case-based reasoning;

• techniques for mapping Bayesian networks to neural network architectures, thus allowing massively parallel implementations of Bayesian reasoning;

• computationally efficient stochastic optimization methods, including a novel variant of simulated annealing with an adaptive cooling schedule.

The theoretical results obtained have been empirically tested by using several real-world data sets. Some of the JAVA software used in these experiments can be downloaded from the group home page (http://www.cs.helsinki.fi/research/cosco/). Examples include

• BAYDA: Bayesian Predictive Discriminant Analysis with feature selection.

• D-SIDE: a Bayesian decision support system with JAVA interface.

In the empirical tests with various public domain classification datasets, D-SIDE consistently outperforms the results obtained by alternative approaches, such as decision trees and neural networks.

Basic research work by the group has been supported by grants from the Academy of Finland, University of Helsinki, and various foundations. More applied work has been performed with support from TEKES and the domestic and foreign industrial partners which include, e.g., Kone, Nokia, ABB, Enso and the AT&T corporations. Some of the resulting software has been adopted in the industry.

Current members of the CoSCo group are Doc. Henry Tirri, Dr. Petri Myllymäki, M.Sc. Tomi Silander, Petri Kontkanen, Jussi Lahtinen and Kimmo Valtonen. The group has also hosted several foreign academic visitors and graduate students, and participates in two EC-funded research networks (the NeuroCOLT working group on neural and computational
2.2 Projects


Publications: [118, 120-128, 192, 193, 217, 223-233, 236, 237].

Animation Aided Problem Solving (AAPS)

Animation is a standard technique in computer-aided instruction. The project aims at applying the methods of algorithm animation in problem solving. The idea is based on the internal similarity of algorithm design and problem solving.

Traditionally animations were coded by hand demanding much time. We have developed two systems for fast generation of animations for algorithms. Eliot works on Linux workstations and Jeliot runs on the World-Wide Web. The speed of generation is one of the key factors of the applicability of the approach.

Besides generation of algorithm animations we have studied how to prepare animations of simple algorithms with the Microsoft Excel spreadsheet program. We are also developing ways to use Excel animations in problem solving. This approach is important because of the wide availability of Excel. In addition to animation we are making tools for creating and managing concept maps in education and problem solving.

Our topics lie in the borderland of computer science and education, and the research is carried out as a joint project with the Department of Teacher Education. Current members of the AAPS group are Doc. Jorma Tarhio (group leader), Assoc. Prof. Veijo Meisalo (Department of Teacher Education), Ph.Lic. Erkki Sutinen, Erkki Rautama, and Tommi Teräsvirta. The group has got funding from the Ministry of Education and University of Helsinki in 1996-98.

Publications: [69, 273, 281, 284-286, 289, 290].

Individual Research

Visualization in Science and Education (Assoc. Prof. Matti Mäkelä): Advanced computer graphics provides means of visualizing scientific data and complicated formal structures and interactions which have earlier been difficult to manage and conceive by human brains. Scientific visualization offers a method to make pictures from the abstract material produced by scientific models and measurements. Thus, the natural human ability to see and think visually is utilized besides the traditional
scientific thinking. The same applies in education, too. One essential problem is to select a proper type of pictures for representing the abstract data and to formalize the transformation from data to pictures. Another problem is the visual literacy, that is the ability to express thoughts by pictures, and the ability to interpret pictures. The scope of this research is to find out practical recommendations to attack these problems. The present activities will include animation tools, and visualization and hypermedia applications for education. As the problem area is highly interdisciplinary there are several contacts within the university, and with the Helsinki University of Technology, the Center for Scientific Computing, and the University of Industrial Arts (Helsinki), among others.

Publications: [269–271, 274–276].

b) Computer Software

Modelling of Concurrency (MOCO)

Distributed systems and parallel programs are notoriously prone to errors caused by subtle differences in the relative execution orderings of the components of the system. In order to avoid such errors, systems should be carefully specified and analyzed. The international research in this area, often called simply concurrency, has been active in the 1980’s and has resulted in numerous specification methods and models for this purpose, e.g. state transition systems, Petri nets, process algebras and temporal logics. Elegant and rich theories supporting these models have made the models even more useful. Results in these theories can be employed in computer based tools for checking and verifying concurrent real life software.

An important class of distributed algorithms consists of computer communication protocols. We have worked with the problems of protocol analysis since 1984. We begun by constructing a reachability analysis tool PROTAN88 designed for protocols specified with an extended state transition model (ESTELLE specification language). In the 1980’s this tool has been tried out by us in analyzing several well-known real-life protocols like X.25, FTAM and X.411/P1. Some of our improvement suggestions were included in the FTAM state tables published by CCITT and ISO.

Lately, our work has been more conceptual and theoretical. Various equivalence concepts have been studied in process algebra contexts (Milners CCS and ISO LOTOS) and the equivalence preserving minimization of labeled state transition systems has been studied. We have been especially interested in divergence preserving behavioural equivalences and preorders.
The analysis of liveness properties is especially important in the context of specification and verification of temporal logic properties of concurrent systems. Temporal logics are a class of logic-based formal languages containing temporal operators, such as ‘sometime’ or ‘always’. Their applications to verification of concurrent systems have been internationally one of the main research directions in the concurrency research for the last twenty years. In our work on temporal logic, we have examined the preservation of temporal logic properties in the construction of reduced models based on divergence-sensitive equivalences and preorders. We have also examined the relations of temporal logics, particularly the so-called mu-calculi or fixed-point temporal logics, with automata on infinite objects.

A recurrent theme in our research is to make analysis and verification feasible for concurrent systems of realistic size. That is why a goal in our research is to understand and support compositional (or modular) specification and verification of concurrent systems. We have been e.g. analyzing and applying the weakest equivalence (CFFD-equivalence) preserving deadlocks and linear next-timeless linear temporal formulae. Our research employs case analyses (often computer communication protocols) to ensure the practical usefulness of the theoretical concepts.

At the university of Helsinki we have developed a computer-based verification tool BIDMIN allowing us to minimize labeled transition systems with respect to divergence preserving bisimilarity and branching bisimilarity as well as the related congruences. Vesa Hellgren has been developing parallel verification algorithms. The algorithms have been implemented in PAVEL (Parallel Verification of LOTOS/LTS) which is a software running in the Cray T3E supercomputer and a network of Linux workstations. We also often use a LOTOS oriented verification tool ARA TOOLS developed at the Technical Research Centre of Finland http://www.tekes.fi/english/.

Current members of the MOCO group are Prof. Martti Tienari (group leader), Doc. Roope Kaivola, M.Sc. Vesa Hellgren, Ph.Lic. Timo Karvi, M.Sc. Päivi Kuuppelomäki and M.Sc. Matti Luukkainen (Nokia Research Centre). As an associate member of the group we have Prof. Antti Valmari (Tampere University of Technology, Software Systems Laboratory) who is the chief designer of ARA TOOLS. Our concurrency group has been participating (1994–97) in the European COST247-project Verification and Validation Methods for Formal Descriptions.

Publications: [18, 98, 100, 103, 106–111].

Open Distributed Computing Environments (ODCE)

The rapid advances in data communication networks have greatly increased the possibilities to make use of the various computational and information services available within reach of a computer network. These services may have different origins, they are implemented on different types of computational platforms, they can be located on sites far apart, and they can be administered independently. As different services become reachable, interoperability of services must be considered. Distributed computation needs an enhanced functionality of the infrastructure, including the emerging technologies.

The research of the ODCE group contributes to open architecture models, such as the Open Distributed Processing reference model (RM-ODP) standardised by ISO and ITU, and open platforms, such as OMG CORBA. The research is also in close relationship with telecommunication related architectures, like TINA.

The department started research activity in the area of open distributed processing at the end of 1980's. In our first project (AHTO) we designed a distributed software environment (middleware) offering for a user a homogeneous interface to the computing services available in a heterogeneous computer network. Since that time the department has participated in the development of the Reference Model for Open Distributed Processing, under the auspices of ISO and ITU-T (former CCITT).

The DRYAD project, 1992-1996, concentrated on heterogeneous environments where autonomously administered systems federate with each other. Federation problems arise from the asynchrony of member system evolution, and from the independent technology choices at member systems (hardware, operating system, middleware, languages, applications). The federation solutions are built on meta-information services, like trading services (global and dynamic repository of service providers), and type repository services (dynamic repository of service types, federation contract schemata, and mappings to local technology solutions). The DRYAD project developed a prototype software package for some of the middleware services - especially for the trading function.

The CORBA-FORTE project (CORBA-Based Framework for Telecommunications) is starting in 1998. The project will concentrate on CORBA technology. It will seek general patterns for improving CORBA system performance. Prospects for improved quality of service and performance are found on two areas. First, additional services can be implemented into the platform itself. Second, applications can exploit the platform services more efficiently when tailored object design patterns are used.
2.2 Projects

The members of the ODCE group are Prof. Martti Tienari, Dr. Timo Alanko, Ph.Lic. Lea Kutvonen, Chief Systems Analyst Petri Kutvonen, M.Sc. Pekka Kähkipuro and M.Sc. Liisa Marttinen.

Publications: [25–37].

Mobile Computing (MOWGLI)

Recent developments in wireless communication and personal computer technology have laid a new foundation for mobile computing. Modern portable computers and wireless connections have created a new platform for distributed information processing where a mobile user has access to various computing services and information stores any time and independently of his or her present location.

The goal of the MOWGLI project is to study, design and test a data communication architecture for a pan-European GSM-based mobile data service, and alike. The environment of an application consists of mobile PC’s which can be connected over a wireless WAN to the Internet. The work in the project is concentrated on the architectural aspects supporting mobility-aware computing. The main issues have been data transport service for constrained wireless links, control of computing in weakly connected and disconnected states, support for legacy software in a mobile environment and design of mobility-aware application software for nomadic users (file transfer, e-mail, WWW), and communication performance in this environment. The experimental research is based on a prototype implementation of the architecture.

The members of the MOWGLI project group are Prof. Martti Tienari, Dr. Timo Alanko, Assoc. Prof. Kimmo Raatikainen, M.Sc. Heikki Helin, M.Sc. Markku Kojo, and M.Sc. Heimo Laamanen. The industrial partners of the project are Nokia Mobile Phones, Nokia Research Centre, Nokia Telecommunications Systems, SSH Communications Security, and Telecom Finland.

Publications: [6–9, 11–14, 19, 38, 45].

Service Machine Development for an Open Long-term Mobile and Fixed Network Environment (DOLMEN)

The EC/ACTS project DOLMEN (ACTS Ref. AC036), 1995–98, demonstrates, assesses and promotes a Service Architecture (called Open Service Architecture for a Mixed fixed and mobile environment – OSAM) that
meets the requirements of open provision of communication services over both fixed and mobile heterogeneous and multi-provider telecommunications networks.

The project’s approach is based upon the RACE Open Service Architecture (OSA) and also progresses results from RACE projects on mobility. The TINA-C current and future developments have been taken into account. In developing OSAM, the parallel work of related ACTS projects are constantly monitored and utilized as appropriate. As in the case of OSA and TINA-C, the DOLMEN approach subscribes to a vision, beyond that currently offered by IN and TMN, of a telecommunications service infrastructure as a large scale, distributed processing environment.

Participants in DOLMEN are Fondazione Ugo Bordoni (Italy), University of Catania (Italy), FINTRACOM S.A. (Greece), National Technical University of Athens (Greece), Personal Communications Services (UK), Research and Consultancy (UK), Sema Group (France), University of Helsinki (Finland), VTT Information Technology (Finland), Koniglijke PTT Nederland N.V. Research (The Netherlands), AT&T Network Systems Nederland B. V. (The Netherlands), and Telecom Finland (Finland).

The current members of the DOLMEN research group are Assoc. Prof. Kimmo Raatikainen (group leader), M.Sc. Stefano Campadello, M.Sc. Oskari Koskimies, M.Sc. Mika Liljeberg and Jarkko Sevanto.

Publications: [15, 16]

Real-Time Object-Based Database Architecture for Intelligent Networks (RODAIN)

Database technologies will, already in the near future, have a dominant role in telecommunication and data communication networks. The information and knowledge needed in network operations and management will be organized as a logical entity. Due to the world-wide nature of communication the only way to achieve the logical uniformity is the interoperability of autonomous databases.

The research project Darfin, Database ARchitecture For Intelligent Networks, 1993-95, examined database architectures that can fulfill the requirements of Intelligent Networks (IN) and Telecommunication Management Networks (TMN). The current ITU-T Recommendations for TMN are based on object-oriented modelling. In addition, the long-term architecture of IN heavily advocates OO-modelling. Therefore, the project focused on object-oriented real-time databases.
The research project RODAIN continues the work done in the Darfin project. In the project we have designed and specified a real-time object-oriented database architecture for Intelligent Networks and implemented a simple prototype. The current objective of the RODAIN project is to extend the prototype to a distributed real-time object-oriented database that can be used in telecommunication.

Research topics:

- real-time transaction processing,
- main memory databases,
- distribution,
- recovery, and
- fault-tolerance.

The industrial partners and financial supporters of the project are Nokia Telecommunications, Solid Information Technology, and Telecom Finland. The RODAIN project is also funded by the Finnish Technology Development Centre (TEKES).


A Channel into Object-Oriented Protocol Design (Kannel)

The constantly growing complexity of distributed and telecommunications applications has made it necessary to develop high-level application-oriented languages and tools. There exists a number of standardized languages and their implementations in the area, but usually they focus just on some rather narrow aspect of the problem. This makes it necessary to use several unrelated languages and environments when developing a nontrivial distributed application.

This research project develops Kannel, an integrated language for the design and implementation of communication protocols. The main features of Kannel are high-level application support, object-orientation, sound protocol refinement, and visual notations. In contrast to conventional languages in the area, Kannel and its programming environment
provide facilities for all the main tasks in a typical protocol development effort.

The first version of Kannel has been implemented. The environment includes a visual Kannel editor, a translator into C++, and a graphical animator/debugger.

The members of the research group are Assoc. Prof. Jukka Paakki (group leader), M.Sc. Kari Granö, and M.Sc. Antti-Pekka Tuovinen.

Publications: [10, 17, 24, 99].


A Framework Editor for Java (FRED)

Reuse is one of the most promising ways to increase the productivity and quality of software development. Unfortunately, reusability has turned out to be rather hard to be reached both in theory and in practice.

Recently, special object-oriented methods have been developed to solve the reusability problem. In this research project we study two of the most promising ones, design patterns and application frameworks. The former captures general and mature design decisions into a reusable form, whereas the latter category provides large-scale code reuse in the form of an extensible implementation of the core functionality of a family of related applications.

The project develops a prototype CASE tool for the development of reusable application frameworks founded on design patterns. The tool makes it possible to instantiate integrated design patterns as code into the framework and to systematically develop a final application from it. The implementation language of the tool, the frameworks, and the applications is Java.

The project is carried out as cooperation between our department (UH) and the Department of Computer Science at the University of Tampere (UT). The project members are Prof. Kai Koskimies (group leader, UT), Assoc. Prof. Jukka Paakki (UH), M.Sc. Markku Hakala (UT), M.Sc. Juha Hautamäki (UT), M.Sc. Jyrki Tuomi (UT), M.Sc. Antti-Pekka Tuovinen (UH), M.Sc. Antti Viljamaa (UH), and M.Sc. Jukka Viljamaa (UH). The main sponsor of the project is TEKES, and the industrial partners are TT Tieto, Stonesoft, Nokia Telecommunications, Novo Group, Sun Microsystems, Major Blue Company, ICL Data, Dycom, Nokia Research Centre, Telecom Finland, Valmet Automation, and Profit.

Publications: [55–57, 62, 63, 67, 68, 70, 73–75].

2.2 Projects

c) Information Systems

Data Mining

Data mining (or knowledge discovery in databases) is a new research area developing methods and systems for extracting interesting and useful information from large sets of data. Data mining methods can be used in a variety of application areas, such as commercial databases, telecommunications, epidemiological data, etc. The area combines techniques from databases, statistics, and machine learning.

The Data Mining research group has developed data mining methods and studied the theory of data mining. The research started in late 1980’s in the context of developing tools for inferring integrity constraints from databases.

We have developed methods for finding recurrent episodes in event sequences, and used these to locate strong rules about the occurrences of events. Clustering methods have also been applied to locate regularities in sequential data. For numerical time-series data we have developed methods that are able to discover similarities in various aspects of potentially related time-series.

Data mining can produce large amounts of new information. We are working on the data mining process as a whole and on the selection of the interesting regularities in particular. In connection with the Document Management group, we have considered these issues in the analysis of text and structure in marked documents.

The group has studied the theory of data mining, e.g., by looking at the relationship of the logical complexity of the discovered sentences and the sample size needed for discovery, and by investigating various frameworks for data mining.

A growing research topic has been the use of Markov chain Monte Carlo methods in data analysis, in particular in the analysis of event data. We develop tools for the automatic analysis of complex statistical models (Bayesian or full probability models), and we model and analyze data with other scientists, e.g., in epidemiology, paleoecology, and archaeology.

The research is done in several projects funded by the Academy of Finland, TEKES, and the European ESPRIT Programme. The group has close cooperation with the Document Management and Machine Learning groups.

The members of the Data Mining group are Prof. Heikki Mannila (group leader), Dr. Helena Ahonen, M.Sc. Oskari Heinonen, Ykä Huhtala, M.Sc. Mika Klemettinen, Karri-Pekka Laakso, Tommi Mononen, M.Sc.


Document Management (DocMan)

Text with a structure is quite common: dictionaries, reference manuals, and annual reports are typical examples. In recent years, research on systems for processing structured documents has flourished. The SGML and ODA standards have further increased the interest in the area. The Document Management (DocMan) Research Group studies the theory and application of such structured documents.

Structured and Intelligent Documents (SID) is an on-going project within the DocMan group that studies and develops methods for attaching intelligent features to structured documents. The purpose of these features is to make the manipulation (storage, retrieval, and assembly) of documents easier. The project started in 1995. SID is part of the Electronic Printing and Publishing program started by the Technology Development Centre of Finland (TEKES). Funding for SID is provided by TEKES and a group of supporting companies.

One of the basic problems in document management is to provide on-demand generation of individualized documents through dynamic document assembly. Document assembly composes new documents from an existing collection of documents. Naturally, document markup and structure contribute to the retrieval and reuse of document fragments.

An intelligent document contains knowledge about itself and its environment. It supports assembly of documents based on inputs given by the user. It is no longer a passive, linear representation of text, but is able to construct itself dynamically. Document assembly is intelligent when it uses application-domain-specific information about the document in addition to the contents and their structure.

The goals of the SID project include (1) defining the information and the knowledge a structured document must contain so that it can work in an active and intelligent way, (2) developing prototype tools for intelligent assembly, and (3) defining a methodology for incorporating intelligence into document material. As a basis for the project we consider structured documents marked up with SGML. The project combines methods and tools from, e.g., structured-document management, information retrieval, pattern matching, data mining, distributed systems, and machine learn-
When dealing with documents in morphologically rich languages like Finnish, also natural language processing is vital to the success of document assembly.

Other ongoing research within the DocMan group includes creating automatically meaningful fragments of long documents, and classifying roles of structured document elements. Former research projects include the RATI project (1988–91) for building a prototype document manipulation system which provides multiple views of a document and the sgrep project (1995) which designed and implemented a search tool for structured documents. Also some results of the VITAL project (1990–95) are usable in this context: one of the tools built in the VITAL project was a general purpose text transformation generator suitable also for structured document transformations.


Transaction Management Support for Cooperative Applications (TRANSCOOP)

Today’s software tools mostly aim at supporting a single user only. For example, consider text editors for writing documents, CAD-tools or software development environments. There are excellent text editors available, but they do not appropriately support multiple authors to work on the same document concurrently.

A major conceptual problem in this framework is to ensure consistency criteria for the data concurrently processed by multiple users. Conventional database technology already provides mechanisms to absolutely guarantee consistency constraints by controlling the concurrent access of different users to shared data. Unfortunately, existing transaction management concepts are not suitable for supporting and controlling cooperation between users, because they are designed to fully isolate users from each other.

The design of cooperative systems includes the description and formal specification of cooperative activities. Such specifications have to be mapped to a cooperative transaction management model in order to enable appropriate database management support at run time. Cooperative transactions are thus intrinsically complex operations that are difficult to
understand completely. A theory is needed for distinct notions of correctness and for correctness-preserving transformations that allow the designer to map specifications onto implementation platforms in a guided way.

The project finished in 1997. The partners of the project were GMD (D), University of Twente (NL) and Technical Research Centre of Finland (VTT) having the University of Helsinki as a subcontractor.


Publications: [149, 152–155, 158].

2.3 Publications

Recent publications of the department are listed below according to the ACM Computing Reviews (CR) classification system. The list contains selected publications in 1994–95 and new publications since January 1996.

General Literature – Introductory and Survey (A.1)


Computer Systems Organization – Computer-Communication Networks – General (C.2.0)


2.3 Publications

Network Architecture and Design (C.2.1)


**Network Protocols (C.2.2)**


**Network Operations (C.2.3)**


2.3 Publications


Distributed Systems (C.2.4)


31. L. Kutvonen: Overview of the DRYAD trading system implementation. In Proc. IFIP/IEEE International Conference on Distributed Platforms: Client/Server and Beyond: DCE, CORBA, ODP and Advanced


Special-Purpose and Application-Based Systems (C.3)

2.3 Publications


**Software – Programming Techniques – Object-Oriented Programming (D.1.5)**


**Logic Programming (D.1.6)**


2.3 Publications

Software Engineering – Tools and Techniques (D.2.2)


Testing and Debugging (D.2.5)

Distribution and Maintenance (D.2.7)


Programming Languages – Language Classifications (D.3.2)


Processors (D.3.4)


Data – Data Storage Representations (E.2)


2.3 Publications


**Data Encryption (E.3)**


**Theory of Computation – Computation by Abstract Devices – Models of Computation (E.1.1)**


**Complexity Classes (E.1.3)**


**Analysis of Algorithms and Problem Complexity – Nonnumerical Algorithms and Problems (F.2.2)**


2.3 Publications

Logics and Meanings of Programs – Specifying and Verifying and Reasoning about Programs (F.3.1)


Mathematical Logic and Formal Languages – Mathematical Logic (F.4.1)


2.3 Publications


Grammars and Other Rewriting Systems (F.4.2)


Formal Languages (F.4.3)


Mathematics of Computing – Numerical Analysis – Optimization (G.1.6)


Discrete Mathematics – Combinatorics (G.2.1)

Probability and Statistics (G.3)


2.3 Publications


Information Systems – Database Management (H.2)


Physical Design (H.2.2)


Languages (H.2.3)


138. G. Grahne, N. Spyrauros and D. Stamate: Semantics and containment of queries with internal and external conjunctions. In Proc. 6th International Conference on Database Theory, ICDT ’97 (eds. F. Afrati and


Systems (H.2.4)


2.3 Publications


**Heterogeneous Databases (H.2.5)**


**Information Storage and Retrieval – Content Analysis and Indexing (H.3.1)**


2.3 Publications


**Information Search and Retrieval (H.3.3)**


**Information Interfaces and Presentation – User Interfaces (H.5.2)**


**Computing Methodologies – Artificial Intelligence (I.2)**


**Applications and Expert Systems (I.2.1)**

Knowledge Representation Formalisms and Methods (I.2.4)


Learning (I.2.6)


2.3 Publications


2.3 Publications


2.3 Publications

**Problem Solving, Control Methods and Search (I.2.8)**


**Image Processing – Enhancement (I.4.3)**


**Pattern Recognition – Clustering (I.5.3)**


**Simulation and Modeling – Simulation Support Systems (I.6.7)**


**Text Processing – Document Preparation (I.7.2)**


253. H. Ahonen, B. Heikkinen, O. Heinonen and M. Klemettinen: Improving the accessibility of SGML documents — A content-


Computer Applications – Physical Sciences and Engineering (J.2)


Life and Medical Sciences (J.3)


Arts and Humanities (J.5)


Computers in Other Systems (J.7)


Computing Milieux – Computers and Education – Computer Uses in Education (K.3.1)


2.3 Publications


Computer and Information Science Education (K.3.2)


2.3 Publications


Computers and Society (K.4)


**Legal Aspects of Computing — Governmental Issues (K.5.2)**

Chapter 3

Faculty

The members of the faculty are introduced by giving their position, e-mail address, WWW home page, research interests, a sample of recent publications, and recent academic activities.

All e-mail addresses are given in the Internet format. In the domain cs.helsinki.fi there is a general address format containing the first and the last name of a person separated with a period, for example hannu.erkio@cs.helsinki.fi for Hannu Erkiö (note the necessary transliteration é→e, ä→a and ö→o).

The publication numbers refer to the classified listing of publications in Section 2.3.

AHONEN, Helena  Ph.D., Research Assistant

• E-mail: Helena.Ahonen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/hahonen/
• Interests: document management, data mining in text, machine learning
• Publications: [167–170, 198–202, 247–254, 295]
• Memberships of program committees: Knowledge Discovery and Data Mining, KDD ’97, Newport Beach, California, USA

ALANKO, Timo  Ph.D., Assistant Professor

• E-mail: Timo.Alanko@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~alanko/
• Interests: distributed operating systems, mobile systems, performance analysis
• Publications: [6–9, 12, 13, 19]

BACK, Ralph-Johan  Ph.D., Docent, Professor (Åbo Akademi)
• E-mail: backrj@abo.fi
• Url: http://www.abo.fi/~backrj/
• Interests: program construction methods and tools, formal methods, programming language semantics, program verification, multiprocessor technology

ELOMAA, Tapio  Ph.D., Assistant Professor
• E-mail: Tapio.Elomaa@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~elomaa/
• Interests: machine learning, robotics, artificial intelligence, algorithms
• Publications: [189, 208–215]
• Memberships of program committees: World Multiconference on Systemics, Cybernetics and Informatics, SCI ’97, Caracas, Venezuela ○ International Conference on Information Systems Analysis and Synthesis, Orlando, Florida, USA, 1998 ○ 6th Scandinavian Conference on Artificial Intelligence, SCAI ’97, Helsinki, Finland (co-chairman)
• Other activities: Referee for Data Mining and Knowledge Discovery, 1997–

ERKIÖ, Hannu  Ph.D., Docent, Assistant Professor
• E-mail: Hannu.Erkio@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~erkio/
• Interests: user interfaces, computer-supported cooperative work
• Publications: [187, 188]

GRAHNE, Gösta  Ph.D., Docent, Assistant Professor (Currently Associate Professor, Concordia University)
• E-mail: grahne@cs.concordia.ca
• Url: http://www.cs.concordia.ca/~faculty/grahne/
• Interests: artificial intelligence, belief revision, data and knowledge bases, knowledge representation, sequence databases
• **Publications:** [134–138, 190, 191]

• **Memberships of program committees:** 6th Scandinavian Conference of Artificial Intelligence, SCAI ’97, Helsinki, Finland, (chair)

**GRANÖ, Kari**  M.Sc., Research Assistant
- **E-mail:** Kari.Grano@cs.helsinki.fi
- **Url:** http://www.cs.helsinki.fi/~grano/
- **Interests:** distributed computing, programming languages
- **Publications:** [10, 17, 24, 99]

**HÄKKINEN, Auvo**  M.Sc., System Administrator
- **E-mail:** Auvo.Hakkinen@cs.helsinki.fi
- **Url:** http://www.cs.helsinki.fi/~ahakkine/
- **Interests:** computer systems organization, operating systems, software engineering,

**HEIKKINEN, Barbara**  M.Sc., Research Assistant
- **E-mail:** Barbara.Hekkinen@cs.helsinki.fi
- **Url:** http://www.cs.helsinki.fi/~bheikkin/
- **Interests:** document management, digital libraries, information retrieval, data mining
- **Publications:** [247–254]

**HEINONEN, Oskari**  M.Sc., Research Assistant
- **E-mail:** Oskari.Heinonen@cs.helsinki.fi
- **Url:** http://www.cs.helsinki.fi/~oheinone/
- **Interests:** document management, data mining, information retrieval
- **Publications:** [167–170, 244, 247–254]

**JÄRVINEN, Pertti**  Ph.D., Docent, Professor (University of Tampere)
- **E-mail:** pj@cs.uta.fi
- **Url:** http://www.uta.fi/~pj/
• **Interests**: information systems, research methodologies, social effects of computing systems, management of computing activities

**KAIVOLA, Roope**  Ph.D., Docent, Assistant Professor

- **E-mail**: Roope.Kaivola@cs.helsinki.fi
- **Url**: http://www.cs.helsinki.fi/~rkaivola/
- **Interests**: concurrency, temporal logic, process algebra, infinitary automata
- **Publications**: [18, 100, 106–111]
- **Other activities**: Referee for *Theoretical Computer Science* ○ SIAM Journal on Computing ○ 9th International Conference on Computer-Aided Verification, CAV ’97, Haifa, Israel ○ Conference on Computer Science Logic, CSL ’97, Aarhus, Denmark ○ 24th International Colloquium on Automata, Languages and Programming, ICALP ’97, Bologna, Italy ○ 6th Scandinavian Conference on Artificial Intelligence, SCAI ’97, Helsinki, Finland ○ Logic in Computer Science, LICS ’96, New Jersey, USA ○ International Conference on Concurrency Theory, CONCUR ’96, Pisa, Italy

**KÄRKKÄINEN, Juha**  M.Sc., Research Assistant

- **E-mail**: Juha.Karkkainen@cs.helsinki.fi
- **Url**: http://www.cs.helsinki.fi/~tpkarkka/
- **Interests**: algorithms and data structures, string algorithms, data mining, computational biology
- **Publications**: [83, 85–88, 162]

**KARVI, Timo**  Ph.Lic., Assistant Professor (acting)

- **E-mail**: Timo.Karvi@cs.helsinki.fi
- **Url**: http://www.cs.helsinki.fi/~karvi/
- **Interests**: formal specification, process algebras, distributed computing

**KEROLA, Teemu**  Ph.D., Assistant Professor

- **E-mail**: Teemu.Kerola@cs.helsinki.fi
- **Url**: http://www.cs.helsinki.fi/~kerola/
• Interests: performance evaluation, software metrics, data visualization, operating systems

KILPELÄINEN, Pekka  Ph.D., Docent, Assistant Professor

• E-mail: Pekka.Kilpelainen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~kilpelai/
• Interests: document management, pattern matching, data mining and knowledge discovery
• Publications: [71, 89, 117, 133, 139, 247–249, 251, 252, 255]
• Memberships of program committees: International Conference on Electronic Publishing, EP ’98, Saint Malo, France

KIVINEN, Jyrki  Ph.D., Docent, Assistant Professor

• E-mail: Jyrki.Kivinen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~jkinen/
• Interests: machine learning, neural networks
• Publications: [218, 220–222]
• Other activities: Coordinator of the Graduate School in Computer Science and Engineering (HeCSE) 1998–

KLEMETTINEN, Mika  M.Sc., Research Assistant

• E-mail: Mika.Klemettinen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~mklemett/
• Interests: data mining, text mining, information retrieval, user interfaces
• Publications: [20–23, 167–170, 175, 176, 250, 253, 254]
• Other activities: Referee for IEEE Transactions on Knowledge and Data
KOJO, Markku   M.Sc., Assistant Professor
    • E-mail: Markku.Kojo@cs.helsinki.fi
    • Url: http://www.cs.helsinki.fi/~kojo/
    • Interests: mobile computing, distributed systems, data communication
    • Publications: [6–9, 11–14, 19, 38]
    • Other activities: Referee for the 2nd International Mobile Computing Conference, 1996, Hsinchu, Taiwan, R.O.C.  
      • IEEE Transactions on Computers, Special Issue on Mobile Computing, 1996

KORPIMIES, Kai   M.Sc., Research Assistant, Assistant Professor (acting)
    • E-mail: Kai.Korpimies@cs.helsinki.fi
    • Url: http://www.cs.helsinki.fi/~korpimie/
    • Interests: information retrieval
    • Publications: [1, 2, 4, 178, 179]

KUJALA, Teija    M.Sc., Amanuensis
    • E-mail: Teija.Kujala@cs.helsinki.fi
    • Url: http://www.cs.helsinki.fi/~tkujala/
    • Interests: computer-aided software engineering, database design

KUTVONEN, Lea    Ph.Lic., Laboratory Engineer, Researcher
    • E-mail: Lea.Kutvonen@cs.helsinki.fi
    • Url: http://www.cs.helsinki.fi/~summanen/
    • Interests: open distributed processing, heterogeneous system architectures
    • Publications: [28–37]
    • Memberships of program committees: 2nd International Working Conference on Distributed Applications and Interoperable Systems, DAIS ’99, Helsinki, Finland, 1999
• Other activities: Finnish delegate in ISO/IEC JTC1/SC21 WG7 on Open Distributed Processing (ODP) standardisation

KUTVONEN, Petri  Chief Systems Analyst
• E-mail: Petri.Kutvonen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~kutvonen/
• Interests: operating systems, distributed systems, open distributed processing (ODP), trading function
• Publications: [37]

KUUPPELOM ÄIKI, Päivi  M.Sc., Research Assistant
• E-mail: Paivi.Kuuppelomaki@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~kuuppelo/
• Interests: modelling of concurrency
• Publications: [293]

LAINE, Harri  Ph.Lic., Assistant Professor
• E-mail: Harri.Laine@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~laine/
• Interests: computer-aided software engineering (CASE), database design
• Publications: [61, 64, 130]

LEMSTR ÖM, Kjell  M.Sc., Research Assistant
• E-mail: Kjell.Lemstrom@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~klemstro/
• Interests: computer graphics, digital image processing, information retrieval, pattern recognition (pattern matching)
• Publications: [180–182, 243]

LINDÉN, Greger  Ph.D., Assistant Professor (acting)
• E-mail: Greger.Linden@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~linden/
• **Interests:** document management, structured documents, structured transformations, SGML

• **Publications:** [71, 72, 247–249, 256]

• **Other activities:** Member of the board of the SGML Users’ Group Finland 1996–

LINNAINMAA, Seppo  Ph.D., Docent, Research Professor (VTT Information Technology)

• **E-mail:** Seppo.Linnainmaa@vtt.fi

• **Url:** http://www.vtt.fi/tte/staff/lis/

• **Interests:** knowledge-based systems, object-oriented software, intelligent operations management

LOKKI, Heikki  Ph.Lic., Assistant Professor

• **E-mail:** Heikki.Lokki@cs.helsinki.fi

• **Url:** http://www.cs.helsinki.fi/~lokki/

• **Interests:** computer science in biological sciences, analytical methods for ringing recoveries

• **Publications:** [263]

• **Other activities:** Referee of EURING ’97 ♦ Coordinator of the Graduate School in Computational Biology, Bioinformatics and Biometry (ComBi) 1998–

MÄKELÄ, Matti  Dr.Techn., Associate Professor

• **E-mail:** Matti.Makela@cs.helsinki.fi

• **Url:** http://www.cs.helsinki.fi/~mamakela/

• **Interests:** numerical methods, mathematical software, computers in education, computer graphics, scientific visualization

• **Publications:** [269–271, 275, 276]

• **Other activities:** International coordinator and referee of ACM SIGNUM Newsletter
MANNILA, Heikki  Ph.D., Professor

- E-mail: Heikki.Mannila@cs.helsinki.fi
- Url: http://www.cs.helsinki.fi/~mannila/
- Interests: data mining, databases, machine learning, algorithms
- Memberships of program committees: 13th European Meeting on Cybernetics and Systems Research, EMCSR ’96, Vienna, Austria o 15th ACM SIGACT-SIGART-SIGMOD Symposium on Principles of Database Systems, PODS ’96, Montréal, Canada o 3rd International Workshop on Principles of Document Processing, PODP ’96, Palo Alto, California, USA o 22nd International Conference on Very Large Databases, VLDB ’96, Mumbai, India o ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, SIGMOD ’96 DMKD, Montréal, Canada, o 1st European Symposium on Principles of Data Mining and Knowledge Discovery, PKDD ’97, Trondheim, Norway o 3rd International Conference on Knowledge Discovery and Data Mining, KDD ’97, Newport Beach, California, USA (co-chair) o 2nd ACM SIGMOD Workshop on Research Issues on Data Mining and Knowledge Discovery, SIGMOD ’97 DMKD, Tucson, Arizona, USA o European Conference on Machine Learning, ECML ’97, Prague, Czech Republic o 9th International Conference on Scientific and Statistical Database Management Systems, SSDBM ’97, Olympia, Washington, USA o 2nd Pacific-Asia Conference on Knowledge Discovery and Data Mining, PAKDD ’98, Melbourne, Australia o 7th ACM SIGACT-SIGART-SIGMOD Symposium on Principles of Database Systems, PODS ’98, Seattle, Washington, USA o 14th International Conference on Data Engineering, ICDE ’98, Orlando, Florida, USA o 4th International Workshop on Principles of Digital Document Processing, PODDP ’98, Saint Malo, France o 2nd European Symposium on Principles of Data Mining and Knowledge Discovery, PKDD ’98, Nantes, France o 15th National Conference on Artificial Intelligence, AAAI ’98, Madison, Wisconsin USA o 4th International Conference on Knowledge Discovery and Data Mining, KDD ’98, New York, USA

- Other activities: Chairman of the board of the Rolf Nevanlinna Institute, 1995– o Reviewer for ESPRIT project ILP2, 1997– o Editor-in-chief of Data Mining and Knowledge Discovery, 1997– o Member of the KDD Steering Committee, 1996– o External reviewer for professorships at the University of Jyväskylä, Rochester University, Uppsala University, Bar-Ilan
MARTTINEN, Liisa  M.Sc., Assistant Professor (acting)
- E-mail: Liisa.Marttinen@cs.helsinki.fi
- Url: http://www.cs.helsinki.fi/~marttine/
- Interests: computer communication, computer networks, network management, open distributed processing
- Publications: [28–30]

MYLLYMÄKI, Petri  Ph.D., Assistant Professor
- E-mail: Petri.Myllymaki@cs.helsinki.fi
- Url: http://www.cs.helsinki.fi/~myllymak/
- Interests: probabilistic modeling, uncertain reasoning, machine learning, case-based reasoning, neural computing
- Publications: [118, 120–124, 192, 193, 217, 223–233, 236, 237]
- Memberships of program committees: 11th Annual Florida Artificial Intelligence International Research Symposium, FLAIRS ’98, special track on uncertain reasoning, Florida, USA ○ 2nd Nordic Workshop on Genetic Algorithms and Their Applications, NWGA ’96, Vaasa, Finland

NIKLANDER, Tiina  M.Sc., Amanuensis
- E-mail: Tiina.Niklander@cs.helsinki.fi
- Url: http://www.cs.helsinki.fi/~niklande/
- Interests: real-time databases, fault-tolerance
- Publications: [40–44, 46, 65, 131, 145–147]

NURMI, Otto  Dr.rer.pol., Assistant Professor
- E-mail: Otto.Nurmi@cs.helsinki.fi
- Interests: databases, algorithms, data structures, computational geometry
NYKÄNEN, Matti  Ph.D., Assistant Professor (acting)
• E-mail: Matti.Nykanen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~mnykanen/
• Interests: query languages for sequence databases, bioinformatics, theory of computation
• Publications: [76, 77]

ORPONEN, Pekka  Ph.D., Docent, Professor (University of Jyväskylä)
• E-mail: orponen@math.jyu.fi
• Url: http://www.math.jyu.fi/~orponen/
• Interests: computational complexity, neural networks, genetic algorithms, cellular automata

PAAKKI, Jukka  Ph.D., Associate Professor
• E-mail: Jukka.Paakki@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~paakki/
• Interests: software engineering, programming languages: design and implementation
• Publications: [10, 17, 24, 66, 67, 70, 99, 160]
• Memberships of program committees: 7th Nordic Workshop on Programming Environment Research, NWPER ’96, Aalborg, Denmark ○ 3rd International Workshop on Automated and Algorithmic Debugging, AADBUG ’97, Linköping, Sweden ○ 9th International Symposium on Programming Languages: Implementations, Logics, and Programs, PLILP ’97, Southampton, UK ○ 5th Symposium on Programming Languages and Software Tools, Jyväskylä, Finland, 1997 (chair) ○ 10th International Symposium on Programming Languages: Implementations, Logics, and Programs, PLILP ’98, Pisa, Italy
• Other activities: Editor of Tietojenkäsittelytiede (Finnish Journal of Computer Science) ○ Columnist of Tietoyhteys (Magazine by the Centre for

**POLLARI-MALMI, Kerttu**  Techn.Lic., Assistant Professor (acting)
- *E-mail:* Kerttu.Pollari-Malmi@cs.helsinki.fi
- *Interests:* databases, concurrency control, B-trees
- *Publications:* [78]

**PUUSTJÄRVI, Juha**  Ph.Lic., Research Assistant
- *E-mail:* Juha.Puustjarvi@cs.helsinki.fi
- *Url:* http://www.cs.helsinki.fi/~puustjar/
- *Interests:* concurrency control, multidatabases, workflows
- *Publications:* [152–155, 158]

**RAATIKAINEN, Kimmo**  Ph.D., Associate Professor
- *E-mail:* Kimmo.Raatikainen@cs.helsinki.fi
- *Interests:* distributed systems, mobile computing, telecommunications, intelligent networks, performance evaluation, real-time databases
- *Publications:* [6–9, 11–16, 19, 38–50, 65, 131, 142, 143, 145–148, 156]
- *Memberships of program committees:* IFIP 1996 Conference on Intelligent Networks, IN ’96, Melbourne, Australia
- *Other activities:* Member of IFIP TC6 Special Interest Group of Intelligent Networks ◦ Editor of Tietojenkäsittelytiede (Finnish Journal of Computer Science)

**RÄIHÄ, Kari-Jouko**  Ph.D., Docent, Professor (University of Tampere)
- *E-mail:* kjr@cs.uta.fi
RONKAINEN, Pirjo  M.Sc., Research Assistant
  • E-mail: Pirjo.Ronkainen@cs.helsinki.fi
  • Url: http://www.cs.helsinki.fi/~ronkaine/
  • Interests: data mining, databases, knowledge bases, hypertext
  • Publications: [20, 91, 172, 175, 176]

SALMENKIVI, Marko  M.Sc., M.Th., Research Assistant
  • E-mail: Marko.Salmenkivi@cs.helsinki.fi
  • Url: http://www.cs.helsinki.fi/~salmenki/
  • Interests: data mining, computational statistics
  • Publications: [245, 246, 258]

SILANDER, Tomi  M.Sc., Research Assistant
  • E-mail: Tomi.Silander@cs.helsinki.fi
  • Url: http://www.cs.helsinki.fi/~tsilande/
  • Interests: probabilistic modeling
  • Publications: [118, 120–124, 126–128, 217, 223–228]

SIPPU, Seppo  Ph.D., Associate Professor
  • E-mail: Seppo.Sippu@cs.helsinki.fi
  • Url: http://www.cs.helsinki.fi/~sippu/
  • Interests: database structures, query processing, logic databases, spatial databases, parsing
  • Publications: [157]

SUTINEN, Erkki  Ph.Lic., Research Assistant
  • E-mail: Erkki.Sutinen@cs.helsinki.fi
  • Url: http://www.cs.helsinki.fi/~sutinen/
  • Interests: algorithms and data structures, string algorithms, algorithm
animation, computer-aided learning

- **Publications**: [69, 86, 90, 92, 93, 266–268, 272, 273, 277–292]

**TAINA, Juha** Ph.Lic, Research Assistant
- **E-mail**: Juha.Taina@cs.helsinki.fi
- **Url**: http://www.cs.helsinki.fi/~taina/
- **Interests**: real-time databases, object orientation, intelligent networks, main memory databases, database models
- **Publications**: [5, 46–54]

**TARHIO, Jorma** Ph.D., Docent, Associate Professor (University of Joensuu)
- **E-mail**: Jorma.Tarhio@cs.joensuu.fi
- **Url**: http://www.cs.joensuu.fi/~tarhio/
- **Interests**: computer science education, string algorithms, animation of algorithms,

**TIENARI, Martti** Ph.D., Professor, Chairman of the Department
- **E-mail**: Martti.Tienari@cs.helsinki.fi
- **Url**: http://www.cs.helsinki.fi/~tienari/
- **Interests**: modeling concurrency, computer networks and distributed systems, programming languages and compilers, management of computing
- **Publications**: [3, 7, 8, 98, 103]
- **Memberships of program committees**: IFIP WG 6.1 International Conference on Distributed Applications and Interoperable Systems, DAIS ’97, Cottbus, Germany
- **Other activities**: Activities in IFIP (International Federation of Information Processing): General Assembly 1987–96, IFIP Trustee 1989–95, Cognizant Officer of TC2 (Software: Theory and Practice) and TC12 (Artificial Intelligence), Member of Publication Committee, Chairman of the Activity Management Board 1992–95, Chairman of the Technical Assembly 1996 ○ Member of the Finnish Academy of Science and Letters ○ Member of the Finnish Academy of Technical Sciences
TIRRI, Henry  Ph.D., Docent, Senior Research Scientist
- **E-mail:** Henry.Tirri@cs.helsinki.fi
- **Url:** http://www.cs.helsinki.fi/~tirri/
- **Interests:** probabilistic modeling, uncertain reasoning, foundations of machine learning, case-based reasoning, neural computing, transaction processing, nonlinear modeling in education
- **Publications:** [72, 118, 120–128, 149, 153–155, 158, 192, 193, 217, 223–233, 236, 237]
- **Memberships of program committees:** 3rd European Conference on Case-Based Reasoning, EWCBR ’96, Lausanne, Switzerland, 1996 ○ Workshop on Case-Based Learning: Beyond Classification and Feature Vectors, Prague, Czech Republic, 1997 ○ 4th European Conference on Case-Based Reasoning, EWCBR ’98, Dublin, Ireland, 1998

TOIVONEN, Hannu  Ph.D., Assistant Professor
- **E-mail:** Hannu.Toivonen@cs.helsinki.fi
- **Url:** http://www.cs.helsinki.fi/~htoivone/
- **Interests:** data mining, data analysis, machine learning, artificial intelligence, stochastic simulation
- **Memberships of program committees:** 3rd International Conference on
Knowledge Discovery and Data Mining, KDD ’97, Newport Beach, California, 1997 ○ 4th International Conference on Knowledge Discovery and Data Mining, KDD ’98, New York, 1998

- Other activities: Referee for Data Mining and Knowledge Discovery ○ Decision Support Systems, Special Issue on Knowledge Discovery and Its Applications to Business Decision Making, 1996

TUOVINEN, Antti-Pekka  M.Sc., Research Assistant

- E-mail: Antti-Pekka.Tuovinen@cs.helsinki.fi
- Url: http://www.cs.helsinki.fi/~aptuovin/
- Interests: visual languages, object-oriented programming, language engineering, software engineering
- Publications: [69, 73–75, 159, 160, 268, 277, 278, 280, 292]

UKKONEN, Esko  Ph.D., Professor

- E-mail: Esko.Ukkonen@cs.helsinki.fi
- Url: http://www.cs.helsinki.fi/~ukkonen/
- Interests: algorithms and data structures, string algorithms, machine learning, computational biology
- Publications: [84, 87, 88, 114–116, 137, 179, 203–206, 259–262]
- Memberships of program committees: 24th International Colloquium on Automata, Languages, and Programming, ICALP ’97, Bologna, Italy ○ 8th Annual Symposium on Combinatorial Pattern Matching, CPM ’97, Aarhus, Denmark ○ 4th South American Workshop on String Processing, WSP ’97, Valparaíso, Chile ○ 1st International Conference on Discovery Science, DS ’98, Fukuoka, Japan ○ 9th International Workshop on Algorithmic Learning Theory, ALT ’98, Otzenhausen, Germany
- Other activities: Editor-in-Chief of the Nordic Journal of Computing, from 1994 ○ Member of the Steering Committee of the Scandinavian Workshop on Algorithm Theory (SWAT), from 1991 ○ External reviewer for professorships at University of California at San Diego, University of California at Davis, University of Colorado at Boulder, the University of Utah, 1996, University of Tampere 1997, and University of Kuopio 1998 ○ Director of the Graduate School in Computational Biology, Bioinformatics, and Biometry (ComBi), 1998– ○ Biotechnology Prize of 1996 awarded by the Alko-Group Ltd
VALMARI, Antti  Dr.Techn., Docent, Associate Professor (Tampere University of Technology)
• E-mail: Antti.Valmari@tut.fi
• Url: http://www.cs.tut.fi/~ava/
• Interests: computer-aided verification and validation of concurrent and distributed systems

VEIJALAINEN, Jari  Dr.-Ing., Docent, Professor, (University of Jyväskylä)
• E-mail: veijalai@jytko.jyu.fi
• Url: http://www.cs.jyu.fi/~veijalai/
• Interests: heterogeneous autonomous distributed systems, heterogeneous transaction processing, transactional workflow systems, mobile computing, verification and validation of concurrent and distributed systems, multimedia databases

VERKAMO, A. Inkeri  Ph.D., Docent, Assistant Professor
• E-mail: Inkeri.Verkamo@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~verkamo/
• Interests: software performance evaluation, software engineering, data mining
• Publications: [72, 161, 164, 167–170]
• Memberships of program committees: 8th International Conference on Software Engineering and Knowledge Engineering, SEKE ’96, Lake Tahoe, Nevada
• Other activities: Coordinator of the Graduate School in Computer Science and Engineering (HeCSE) 1995–7

VIHAVAINEN, Juha  Ph.Lic., Assistant Professor
• E-mail: Juha.Vihavainen@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~vihavain/
• Interests: principles and implementation of programming languages, language processors, programming paradigms, multi-paradigm programming, object-oriented programming

VILO, Jaak  M.Sc. (University of Tartu), Research Assistant
• E-mail: Jaak.Vilo@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~vilo/
• Interests: computational biology, pattern discovery, machine learning
• Publications: [203–206, 294]

WIKLA, Arto  M.Sc., Assistant Professor (acting)
• E-mail: arto.wikla@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~wikla/
• Interests: computer science education, WWW, programming languages, Java

YLI-HARJA, Olli  Dr.Techn., Assistant Professor (acting)
• E-mail: Olli-Yli-Harja@cs.helsinki.fi
• Url: http://www.cs.helsinki.fi/~yliharja/
• Interests: image processing, nonlinear signal processing, median filters, algorithm analysis, computer-aided sound and music analysis, data structures
• Publications: [119, 129]
Chapter 4

Education

4.1 Educational Program

The students of the department normally start their university studies at the age of 19. Their goal is to receive a B.Sc. (Bachelor of Science) or M.Sc. (Master of Science) degree in computer science requiring three to four or five years of full time study. Beyond the first degree there are two alternative graduate degrees: Ph.Lic. (Licentiate of Philosophy) and Ph.D. (Doctor of Philosophy). The academic year has two semesters: the fall semester lasts from 1 September to 20 December (classes from 11 September to 10 December), while the spring semester lasts from 16 January to 31 May (classes from 16 January to 10 May, excluding one week of Easter vacation). It is also possible to study in summer. Intensive courses of 4–5 weeks covering introductory topics are offered in June and August. Graduate courses are also organized in cooperation with other Finnish universities during the summer. These courses typically last for one week and are intended for Ph.Lic. and Ph.D. students. These are often given in English by foreign visitors.

In order to obtain a B.Sc. degree a student must earn 120 units of academic credit. For a M.Sc. degree, 160 units of credit as well as a thesis is required. One credit should normally correspond to roughly one week (40 hours) of study. Our students typically register for 12 credits (“study weeks”) in the fall semester and 15 credits in the spring semester. During the summer sessions a student can earn an additional 8–10 credits. Most students, however, work in industry during the summer to gain practical experience in data processing and to improve their financial situation. This is actually what the department recommends. Thus, a normal student should earn 27 credits a year, an exceptionally diligent full-year student 40 credits. Nevertheless, there is a considerable variation in study
efficiency among students.

Our typical course consists of about 50 to 60 lectures (a lecture lasts 45 minutes) and of about 20 to 30 hours of problem solving, discussion and repetition sessions in small groups of about 10 to 20 students. Each course is examined individually with grades: 3/3 = excellent, 2/3 = good, 1/3 = satisfactory. A typical course is worth 4 or 5 credits. The computer laboratory is supervised in small groups of 6 to 12 students. Students also attend seminar courses, the enrollment of which ranges from 5 to 15 students. In these seminars the students read current literature, write essays and give oral presentations. A seminar group normally meets 2 hours per week yielding 2 credits per semester.

In order to receive a M.Sc. degree in computer science, students are required to earn their credits as follows:

<table>
<thead>
<tr>
<th>Course</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Computer science</td>
<td>≥ 95 cr</td>
</tr>
<tr>
<td>Mathematics</td>
<td>≥ 26 cr</td>
</tr>
<tr>
<td>Physics</td>
<td>≥ 15 cr</td>
</tr>
<tr>
<td>General studies</td>
<td>9 cr</td>
</tr>
<tr>
<td>Total</td>
<td>≥ 160 cr</td>
</tr>
</tbody>
</table>

In mathematics, the obligatory courses are calculus (11 cr), algebra (5 cr), logic (5 cr), and probability (5 cr). Physics can be replaced with almost any other subject, such as economics, administration, statistics, or psychology. For a B.Sc. degree, 55 credit units of computer science is sufficient.

The computer science studies for a M.Sc. degree can be subdivided as follows:

<table>
<thead>
<tr>
<th>Course</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Obligatory courses and laboratories</td>
<td>35 cr</td>
</tr>
<tr>
<td>Elective courses</td>
<td>≥ 28 cr</td>
</tr>
<tr>
<td>Seminars</td>
<td>≥ 4 cr</td>
</tr>
<tr>
<td>Project work</td>
<td>8 cr</td>
</tr>
<tr>
<td>M.Sc. Thesis, Scientific writing</td>
<td>20 cr</td>
</tr>
<tr>
<td>total</td>
<td>≥ 95 cr</td>
</tr>
</tbody>
</table>

The obligatory computer science courses and laboratory work currently cover (academic year 1997–98) the following areas:
4.1 Educational Program

<table>
<thead>
<tr>
<th>Course</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>Introductory programming (Java)</td>
<td>6 cr</td>
</tr>
<tr>
<td>Data structures</td>
<td>6 cr</td>
</tr>
<tr>
<td>Operating systems and hardware architecture</td>
<td>7 cr</td>
</tr>
<tr>
<td>Information systems and databases</td>
<td>8 cr</td>
</tr>
<tr>
<td>Theory of computation</td>
<td>4 cr</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>31 cr</strong></td>
</tr>
</tbody>
</table>

In principle, students are fairly free to choose any elective courses. They normally follow the recommendation of the department by building up a specialized background knowledge for a successful thesis in one of our research groups. Thus, a student might specialize according to his/her study goals, interests and talents, towards, e.g., theoretical computer science, information systems, telecommunications software, distributed systems, operating systems, artificial intelligence, or software engineering.

To start studies for the postgraduate degrees Ph.Lic. (Licentiate of Philosophy) and Ph.D. (Doctor of Philosophy) in Computer Science, a student having shown good academic standing in his/her M.Sc. studies contacts a professor of the department. At first, a personal study program is designed for the student. It outlines the field of specialization of the studies, the topic for the thesis, and the content and the schedule of the coursework. Each student is assigned a personal advisor. See Section 4.2 (Graduate Schools) for more details.

The requirements for the Ph.Lic. degree can be summarized as follows:

<table>
<thead>
<tr>
<th>Elective courses and seminars</th>
<th>Credits</th>
</tr>
</thead>
<tbody>
<tr>
<td>in computer science</td>
<td>20 cr</td>
</tr>
<tr>
<td>in mathematics</td>
<td>20 cr</td>
</tr>
<tr>
<td>Ph.Lic. thesis</td>
<td>50 cr</td>
</tr>
<tr>
<td><strong>Total</strong></td>
<td><strong>90 cr</strong></td>
</tr>
</tbody>
</table>

The elective courses in mathematics can be replaced with coursework in other subjects such as physics, economy, psychology, etc., or additional courses in computer science. The Ph.Lic. thesis is normally written in English. The allocation of credits for thesis research indicates that after the required coursework it should take 1–2 years to prepare a Ph.Lic. thesis.

It is important that the student takes the courses and the seminars early enough to obtain sufficient background for writing the thesis. Active participation in seminar courses is particularly useful as is attending international schools and specialized research courses. Such courses are also regularly given at the department.
The requirements for the Ph.D. degree are otherwise the same as for the Ph.Lic. degree, but a Ph.D. thesis demands more work, from 2 to 3 years, roughly one year more than a Ph.Lic. thesis. The Ph.D. degree can be achieved directly, although we often recommend that our students take the Ph.Lic. degree first, and then by improving and extending their Ph.Lic. research, achieve the Ph.D. level.

The Ph.D. theses are written in English. A thesis should include a scientific contribution which is significant enough to be publishable internationally. A Ph.D. thesis (as well as a Ph.Lic. thesis) can also be assembled from a number of published articles or congress papers, possibly written jointly with other authors. A dissertation of this type, which is actually fairly common, consists of an introductory survey written by the candidate alone, with the individual articles as appendices.

Preparing the thesis is clearly the most demanding part of the Ph.D. and Ph.Lic. studies. To succeed with the thesis it is recommendable that a student works within a research group at the department. The support and the criticism given by the group is often essential for making progress in the work.

4.2 Graduate Schools

The department is at the moment involved in two separate graduate schools, the Helsinki Graduate School in Computer Science and Engineering (HeCSE) and the Graduate School in Computational Biology, Bioinformatics, and Biometry (ComBi).

4.2.1 HeCSE

The Helsinki Graduate School in Computer Science and Engineering (HeCSE) is a postgraduate program in computer science and computer engineering jointly offered by the Helsinki University of Technology and the University of Helsinki. In addition to the Department of Computer Science at the University of Helsinki, the following laboratories of the University of Technology participate in HeCSE:

- Laboratory of Information Processing Science (Prof. Martti Mäntylä),
- Laboratory of Telecommunications Software and Multimedia (Prof. Arto Karila),
- Digital Systems Laboratory (Prof. Leo Ojala),
Laboratory of Signal Processing and Computer Technology (Prof. Iiro Hartimo),

Laboratory of Computer and Information Science (Prof. Erkki Oja),

and

Neural Networks Research Centre (Prof. Teuvo Kohonen).

The main areas of HeCSE are:

- **Software Systems** (information technology for production, multimedia, database systems, data structures),
- **Software Engineering** (embedded systems, specification methods),
- **Telecommunication Software and Distributed Systems** (formal methods, data communication software, signal processing), and
- **Learning and Intelligent Systems** (pattern recognition, neural networks, machine learning, knowledge based systems).

HeCSE provides a program aiming at a doctoral degree in four years on the basis of a master’s degree in computer science or a related field. The student is appointed a supervising professor, with whom a study plan is made. The study plan contains an individual timetable. The student participates in the work of a research group. The progress of the student is followed by his/her supervisor and at seminars. Additionally, the student must report on his/her work twice a year. The student is formally enrolled at the university of his/her supervising professor.

The studies consist of courses and seminars (20–25%) and research work ending in a thesis (75–80%). There is a compulsory course on Writing Scientific English and five courses on the main areas of HeCSE, three of which are recommended to be taken by the student. The other courses are chosen individually. The student takes the courses and most of the seminars during the first two years. At the end of the second year, the student presents a thesis proposal to the board of HeCSE. The doctoral thesis is expected to be finished by the end of the fourth year. The studies end with a public doctoral dissertation.

HeCSE has in all 25 graduate assistantships, providing funding for about half of the students of the graduate school. Other sources of funding are employment in research projects as research assistants or in the participating laboratories as part time teachers. During the first three years of its existence (1995–1997), HeCSE has produced 14 doctors (7 of these at
the University of Helsinki) and 13 licentiates (3 of these at the University of Helsinki), with the number steadily growing by the year.

The Finnish graduate schools in computer science (Helsinki, Jyväskylä, Turku, Tampere and Eastern Finland) co-operate. Joint courses are arranged and students of one graduate school can participate in the courses arranged by another graduate school. Also participation in some courses arranged by industry is possible.

4.2.2 ComBi

The Graduate School in Computational Biology, Bioinformatics, and Biometry (ComBi) is a new postgraduate program jointly offered by the University of Helsinki, the University of Turku and the Center for Scientific Computing of Finland.

The research goal of ComBi is to develop computational, mathematical, and statistical methods and models for biological sciences. To that end, ComBi will educate Ph.D.’s with high-quality methodological expertise. In their thesis work the students are expected to apply this expertise to computational, data analysis, or modeling problems in biology or in some related field. The thesis projects are carried out in close cooperation with one or more research groups in the application area.

Computational biology is a new field of research which develops models and software implementations for computational problems in molecular biology, biotechnology, and genetics. Such means are needed both in the basic research and in the industrial applications of biotechnology. Computer-aided DNA sequencing, sequence comparison, prediction of protein structures, docking of molecules, and the interpretation of electron microscopy or NMR data are examples of typical computationally intensive tasks. Bioinformatics refers to the development and use of (molecular) biological databases.

Biometry investigates statistical modeling problems related for example to genetic mapping (linkage analysis), to the genetic and environmental risk factors of complex diseases, or to the spreading of infectious diseases. Further topics include population dynamics and numerical taxonomy with applications to ecology, microbiology, and genetics. Management of spatial data (GIS) is also often needed.

ComBi incorporates the following departments and institutions that provide the methodological education of the school:

- Department of Computer Science, University of Helsinki
4.2 Graduate Schools

- Prof. Heikki Mannila (data mining, databases, computational data-analysis)
- Prof. Esko Ukkonen (computational biology, bioinformatics, pattern matching, machine learning),
- Rolf Nevanlinna Institute, University of Helsinki
  - Prof. Elja Arjas (statistics, especially biometry),
- Department of Mathematical Sciences, University of Turku
  - Prof. Mats Gyllenberg (population dynamics, numerical taxonomy), and
- Center for Scientific Computing
  - Prof. Olle Teleman (molecular modeling, structural biology, scientific computing).

The application oriented part of the thesis work of the students of ComBi will be done in cooperation with suitable research groups working at the universities mentioned above or at other institutions.

The school is coordinated by the Department of Computer Science of the University of Helsinki. Professor Esko Ukkonen is the Director of ComBi.

ComBi provides a program aiming at a doctoral degree in four years on the basis of a master’s degree. The master’s degree should preferably be in computer science, (applied) mathematics, or statistics. However, a degree in natural sciences (e.g., biological sciences, biochemistry, chemistry, physics, biotechnology) or in engineering is possible, provided that the student has sufficient knowledge of computational and mathematical methods.

The student is appointed a supervising professor, with whom a personal study plan is made. A central principle of the school is that a major part of the thesis work is carried out by participating in the research work in some biological application area. The student is formally enrolled at the university of his/her supervising professor.

The studies consist of courses and seminars (about 20 %) and research work leading to a Ph.D. thesis. The student should take at least three of the five general courses organized by the school. The rest of the courses should be selected individually according to the special field of the thesis project. Most of the courses and seminars are arranged during the normal
academic semesters (January to May, September to December), but some intensive courses are also planned during summer and winter breaks. Moreover, the school aims at organizing and financing working periods abroad for its students.

The progress of the student is followed by his/her supervisor and at the joint seminars arranged by the school. The student must report on his/her progress twice a year. The student should present a thesis proposal by the end of the second year, and the thesis is expected to be finished by the end of the fourth year. The studies end with a public defense of the thesis.

4.3 Course Descriptions

Undergraduate Courses 1996–98


Introduction to Programming (4 cu)  The course provides the student with the basic principles of programming: algorithms, programming techniques, and object-orientation. Java is the language used, but all of its features are not covered.


Information Systems (4 cu)  Principles of relational databases, SQL, database programming, application development, databases and the WWW, user interfaces, information system development, object-oriented analysis, use case model, introduction to database design.

Programming Project (2 cu)  The student designs, documents and programs a complete, realistic program. In the course of the development she/he also gives small lectures and demonstrations about the project.
Data Structures (4 cu)  Basic data structures. Applications to algorithms. Analysis of algorithms. Implementations of data structures and algorithms. Memory management.

Concurrent Systems (4 cu)  Structure and implementation of concurrent and distributed systems. The main emphasis is on solving problems in operating systems.


Data Structures Project (2 cu)  A simulator or some other fairly large program is designed, programmed, tested and documented.

Information Systems Project (2 cu)  A small ADP-system is designed and programmed.


Languages for Artificial Intelligence (3 cu)  Fundamentals of declarative and symbolic programming. LISP and PROLOG basics. Hands-on experimentations and implementations.


Semantics of Programs (3 cu)  Axiomatic semantics of programs. Weakest precondition calculus for the guarded command language of Dijkstra. Development of small programs based on the programming logic.


Programming Techniques (C++) (3 cu)  Abstract data types, class libraries, generic programming, object-oriented programming.

Programming Techniques (Ada) (3 cu)  Basic Ada data and control structures, packages, concurrency and object-oriented programming. Applicability of Ada in the implementation of a real-time system.

Software Engineering Laboratory (6 cu) Each student takes part in a project where a group of students analyzes the requirements of a software product, designs, implements, and tests the product, using systematic software engineering methods and tools. The group assignment may also be focused on some sub-phase of the software life-cycle, such as evolution or maintenance of an existing software system.

UNIX Principles (1 cu) Principles of the UNIX environment for end users, includes principles of file system, shell, wildcards, protection, I/O, text editing, regular expressions, sorting and searching, awk, program development, project maintenance, and networking commands.

UNIX Platform (3 cu) The programming interface to the UNIX system: system calls and library functions for process control, memory management, file systems and peripherals, tools for interprocess communication.

Network Programming (4 cu) Internet networking services provided to application programs and TCP/IP protocols, including design and implementation principles of these protocols. The focus is on practical aspects of designing and implementing distributed applications using these protocols.

Wireless Data Communication Networks (1 cu) Basics of wireless data communication. GSM, wireless LANs, cellular telephone networks, packet radio networks, wireless LANs, satellites and UMTS.


Introduction to Document Preparation (2 cu) HTML and WWW techniques, document preparation and electronic publishing.
Scientific Writing (4 cu)  Sources of scientific information. Use of libraries and scientific data bases. The structure and details of a scientific publication. Examples of scientific Finnish or Swedish. Three individual survey writing exercises.

Tutoring (1 cu)  In tutoring the second year students are divided into groups of ca. 15 students. Each group is led by a teacher or researcher (tutor) and the group meets monthly during two academic years. In the meetings students introduce matters related to their studies for discussion. In addition each student meets regularly with his/her tutor for private discussions to solve problems in his/her studies. The objective of tutoring is to reduce the number of student dropping out, uncover bottlenecks in the studies, and improve the relationships between students and teachers.

Graduate Courses 1996–98

a) General Computer Science


4.3 Course Descriptions

**Problem Solving (2 cu)**  Creative problem solving in learning, teaching, and research.


**Advanced Computer Graphics (4 cu)**  A selection of advanced topics such as ray tracing, radiosity, solid modeling, illumination and color, scientific visualization, etc. are taken as a theme of the course. Individual and group work, report writing and oral presentations by the participants.

**Fundamentals of Image Processing (2 cu)**  Basic principles and methods of digital image processing.

**Applications of Image Processing (2 cu)**  How to use digital image processing in various applications, possibilities and restrictions.


b) Computer Software


**Computer Networks (4 cu)**  Formal specification and analysis of communication protocols, specification case studies. Security in computer networks: encryption, authentication, digital signatures etc.
Temporal Logic and Verification (3 cu)  An introduction to linear and branching-time temporal and modal logics and their applications in specifying and verifying requirements of concurrent and reactive systems.

Principles of Concurrent Programming I (3 cu)  Basic abstractions in concurrent programming, analysis and verification of algorithms and synchronization primitives for shared-memory parallel programs.

Principles of Concurrent Programming II (3 cu)  An overview of concurrent programming in a distributed memory computer architecture where communication is realized with message passing. Besides formal analysis of this computing model a collection of distributed algorithms are studied to illustrate and concretize this approach.

Computer Architectures (4 cu)  Structure of computer architectures, from instruction sets to I/O systems. The main emphasis is on uniprocessor systems.


Compilers (6 cu)  Lexical analysis, syntax analysis, semantic analysis, and code generation; use of metatools, laboratory assignments.

Mobile (Wireless) Data Communications (3 cu)  Basic technologies, major challenges, and future of wireless data networking including: wireless data networks (GSM, Mobitex, CDPD, etc.), challenges of mobile (wireless) data networking, mobile IP, current solutions to the challenges, and possible applications for nomads. The course includes also minor projects in which participants design systems that employ wireless data networking.

4.3 Course Descriptions

c) Information Systems


**User Interfaces (4 cu)**  Concrete user interface solutions and their foundations. Aspects of user interface design: cognitive psychology, graphical user interfaces, user interface management systems, usability and testing. Current fields of research, e.g., WWW, multimedia, computer-supported co-operative work, virtual reality.

Information Retrieval Methods (3 cu)  Traditional and new information retrieval methods, including networked retrieval and information exploration. Information filtering. Digital libraries.


Knowledge Discovery in Databases (3 cu)  Data mining. Finding frequent patterns in sequences. Integrity constraints in databases. Advanced techniques.

Seminar Courses in 1996–98

a) General Computer Science

- Analogic Computation
- Animation of Algorithms
- Artificial Life
- Biodatabases
- Computers in Special Needs Education
- Computer Science Education
- Computer Uses in Education
- Evolutionary Computation
- Information Technology as a Means for Communication
- Quantum Computing
- Reasoning About Knowledge
- Research Seminar on Computational Biology
- Research Seminar on Machine Learning and Data Mining

b) Computer Software

- Computer Network Architectures
- ATM Networking
- CASE Systems
- Colloquium on Distributed Systems
- The CORBA Platform for Distributed Software Systems
- Data Communications Technology
- Security in Distributed Systems
4.4 Accepted Theses

Design and Implementation of the Linux Operating System
Implementation of Object-Oriented Languages
Internet Protocols
Object Architectures
Object-Oriented Analysis and Design
Object-Oriented Programming
Open Distributed Computing
Real Time Systems
Research Seminar on Formal Specification and Verification
Research Seminar on Mobile Workstations
Software Engineering
Software Metrics
Software Testing
Supercomputers with Distributed Memory
Telecommunications Technology
TINA
The Z Specification Language

c) Information Systems

Geographical Information Systems
Hypertext Systems
Information Retrieval Methods
Knowledge Discovery
Knowledge Discovery and Pattern Recognition
Object-Oriented Databases
Research on User Interfaces
Research Seminar on Databases
Research Seminar on Information Systems
Text Databases
User Interface Research

4.4 Accepted Theses

Theses for Doctor of Philosophy


4.4 Accepted Theses


Hannu Toivonen: Discovery of Frequent Patterns in Large Data Collections. A-1996-5.


Selection of Theses for Licentiate of Philosophy


Lea Kutvonen: The Role of Trading Function in Open Distributed Processing Infrastructure. C-1996-84.


Selection of Master’s Theses in 1996–1998

a) General Computer Science

1996


1Written mostly in Finnish, some in Swedish or English.
4.4 Accepted Theses


Hanna Keskinen: The Computer Driving License Examination and Its Relationship to the Basic Computer Science Studies at Universities. C-1996-100.


Mika Seppänen: Constructing Bayesian Networks from Data. C-1996-33.
Mika Sorsa: Qualitative Bayesian Networks. C-1996-98.

1997

Teemu Antti-Poika: Using Bayesian Networks in Classification. C-1997-68.


Jan Täppinen: Neural Networks in Financial Applications. C-1997-76.

b) Computer Software

1996


1997


Mika Hatanpää: Software Reuse. C-1997-64.


Esko Koskimaa: Meta Facilities in Object-Oriented Languages. C-1997-86.


Timo Suominen: Analysis of Embedded SQL in the HyperSoft System. C-1997-44.


Katri Turunen: A Distributed Memory-File System Based on the Virtual Memory Concept. C-1997-77.


c) Information Systems

1996


Marko Salmenkivi: Modeling Event Series and Generating MCMC-Simulation Programs to Approximate the Posterior Distribution of Model Parameters. C-1996-70.

1997

Harri Rautio: An Implementation of Particle Systems in a PC. C-1997-42.
4.5 Abstracts of Recent Ph.D. Theses


Dictionaries, user manuals, encyclopedias, and annual reports are typical examples of structured documents. Structured documents have an internal, usually hierarchical, organization that can be used, for instance, to help in retrieving information from the documents and in transforming documents into another form. The document structure is typically represented by a context-free or regular grammar. Many structured documents, however, lack the grammar: the structure of individual documents is known but the general structure of the document class is not available. Examples of this kind of documents include documents that have Standard Generalized Markup Language (SGML) tags but not a Document Type Definition (DTD).

In this thesis we present a technique for generating a grammar describing the structure of a given structured document instance. The technique is based on ideas from machine learning. It forms first finite-state automata describing the given instances completely. These automata are modified by considering certain context conditions; the modifications correspond to generalizing the underlying language. Finally, the automata are converted into regular expressions, which are then used to construct the grammar. Some refining operations are also presented that are necessary for generating a grammar for a large and complicated document. The technique has been implemented and it has been experimented using several document types.

Hannu Toivonen: Discovery of Frequent Patterns in Large Data Collections. A-1996-5.

Data mining, or knowledge discovery in databases, aims at finding useful regularities in large data sets. Interest in the field is motivated by the growth of computerized data collections and by the high potential value of patterns discovered in those collections. For instance, bar code readers at supermarkets produce extensive amounts of data about purchases. An analysis of this data can reveal useful information about the shopping behavior of the customers. Association rules, for instance, are a class of patterns that tell which products tend to be purchased together.

The general data mining task we consider is the following: given a class of patterns that possibly have occurrences in a given data collection, determine which patterns occur frequently and are thus probably the most
useful ones. It is characteristic for data mining applications to deal with high volumes of both data and patterns.

We address the algorithmic problems of determining efficiently which patterns are frequent in the given data. Our contributions are new algorithms, analyses of problems, and pattern classes for data mining. We also present extensive experimental results. We start by giving an efficient method for the discovery of all frequent association rules, a well known data mining problem. We then introduce the problem of discovering frequent patterns in general, and show how the association rule algorithm can be extended to cover this problem. We analyze the problem complexity and derive a lower bound for the number of queries in a simple but realistic model. We then show how sampling can be used in the discovery of exact association rules, and we give algorithms that are efficient especially in terms of the amount of database processing. We also show that association rules with negation and disjunction can be approximated efficiently. Finally, we define episodes, a class of patterns in event sequences such as alarm logs. An episode is a combination of event types that occur often close to each other. We give methods for the discovery of all frequent episodes in a given event sequence.

The algorithm for the discovery of association rules has been used in commercial data mining products, the episode algorithms are used by telecommunication operators, and discovered episodes are used in alarm handling systems.

**Henry Tirri: Plausible Prediction by Bayesian Inference. A-1997-1.**

The capability to perform inference with uncertain and incomplete information is characteristic to intelligent systems. Many of the research issues in artificial intelligence and computational intelligence can actually be viewed as topics in the “science of uncertainty,” which addresses the problem of plausible inference, i.e., optimal processing of incomplete information. The various different approaches to model and implement intelligent behavior such as neural networks, fuzzy logic, non-monotonic (default) logics and Bayesian networks all address the same problem of finding an appropriate language and inference mechanism to perform plausible inference, needed to implement such activities as prediction, decision making, and planning.

In this work we study the problem of plausible prediction, i.e., the problem of building predictive models from data in the presence of uncertainty. Our approach to this problem is based on the language of Bayesian probability theory both in its traditional and information the-
oretic form. We study Bayesian prediction theoretically and empirically with finite mixture models. Such models are interesting due to their ability to accurately model complex distributions with few parameters. In addition, finite mixture models can be viewed as a probabilistic formulation of many model families commonly used in machine learning and computational intelligence. We first address the question of how an intelligent system should predict given the available information. We present three alternatives for probabilistic prediction: single model based prediction, evidence based prediction, and minimum encoding based prediction. We then compare the empirical performance of these alternatives by using a class of finite mixture models. The empirical results demonstrate that, especially for small data sets, both the evidence and the minimum encoding approaches outperform the traditionally used single model approach.

We then focus on the problem of constructing finite mixture models from the given data and a priori information. We give the Bayesian solution for inferring both the most probable finite mixture model structure, i.e., the proper number of mixture components, and the most probable model within the class. For general mixture models the exact solution in both problems is computationally infeasible. Thus we also evaluate the quality of approximate approaches.

The Bayesian predictive approach presented can be applied to a wide class of prediction problems appearing in various application domains, e.g., medical and fault diagnostic problems, design problems and sales support systems. Using publicly available data sets, we demonstrate empirically that Bayesian prediction with finite mixtures is highly competitive when compared to the results achieved with other popular non-Bayesian approaches using, for example, neural network and decision tree models. The Bayesian prediction method presented constitutes the kernel of the D-SIDE/C-SIDE software currently used in industrial applications.


We present two techniques for transforming structured documents. The first technique, called TT-grammars, is based on earlier work by Keller et al., and has been extended to fit structured documents. TT-grammars assure that the constructed transformation will produce only syntactically correct output even if the source and target representations may be specified with two unrelated context-free grammars. We present a transformation generator called ALCHEMIST which is based on TT-grammars. ALCHEMIST has been extended with semantic actions in order to make it possible to build full scale transformations. ALCHEMIST has been extens-
ively used in a large software project for building a bridge between two development environments. The second technique is a tree transformation method especially targeted at SGML documents. The technique employs a transformation language called TranSID, which is a declarative, high-level tree transformation language. TranSID does not require the user to specify a grammar for the target representation but instead gives full programming power for arbitrary tree modifications. Both ALCHEMIST and TranSID are fully operational on UNIX platforms.


New application areas for databases demand the possibility of managing not only the traditional atomic but also structured data types. One type of this kind is a finite sequence of characters drawn from a finite alphabet. These string databases are important for example in molecular biology data management, because they can for instance represent DNA sequences directly as strings from alphabet \(\{A, C, G, T\}\). Then the query language must be able to manipulate strings both as indivisible entities and as ordered sequences of distinct characters, perform pattern matching in strings, compare several strings with each other, and generate new strings not yet in the database. This work presents Alignment Calculus, a new modal logic extension of the relational calculus, which satisfies all these requirements with a new unified string manipulation formalism. This formalism is based on the concept of multiple alignment of several strings; computational molecular biology employs an analogous concept. In the language, alignments are described logically with string formulae, a new extension of linear-time temporal logic into multiple strings. The abstract expressive power of Alignment Calculus is shown to equal the Arithmetical Hierarchy. The work develops also a syntactically safe sublanguage, whose queries require only finite computational resources. This sublanguage is constructed by first identifying a decidable subcase of string formula safety analysis, even though the general problem is shown to be undecidable. This safe sublanguage is shown to equal the Polynomial-time Hierarchy in its expressive power, and therefore to capture all the string queries occurring in practical situations. The operational counterpart of Alignment Calculus, Alignment Algebra, is developed by replacing the selection operator of relational algebra with one employing multitape non-deterministic finite state automata corresponding to the aforementioned string formulae, and adding an explicit domain symbol. The aforementioned safe sublanguage has also a counterpart in this algebra: expressions, in which all the domain symbol occurrences are restricted by imme-
A finite evaluation strategy is developed for 
diately enclosing automata. A finite evaluation strategy is developed for 
these expressions.
Chapter 5

Library

The department maintains a library with large collections of literature in computer science. The library is jointly financed with the University Computing Centre and is mainly used by the staff and advanced students of the department.

Established in 1967, the library now holds about 52,000 volumes of literature, making it the largest computer science library in Finland. The annual cumulation is about 1,200 monographic titles and 300 journal subscriptions. The floor area is 408 sq. meters including a reading room of 60 seats. Admission to the premises is free and collections are freely available to all visitors. Home loans, however, are normally granted only to university personnel and advanced students of the department (for exceptions, consult the library staff).

To help users search and locate the required literature, the library maintains a www-browsable database of its holdings (http://www.cs.helsinki.fi/kirjasto/). The database includes all journal titles and about 37,000 monographic titles, classified according to the CR Classification System of the ACM. The library is also responsible for the distribution of departmental reports, including Ph.D. theses. Paper copies may be requested from the library, and electronic versions are accessible through the department’s FTP server.

The library has two full-time employees, one librarian and one secretary, assisted in their work by a library committee comprising of several members of the faculty. Any inquiries to the library are welcome by post, telex, telephone or e-mail (library@cs.helsinki.fi).
Chapter 6

Computing Facilities

The department offers a wide range of services to support computing activities of the academic staff and students. The policy is to provide access to advanced hardware and software systems.

The computing facilities include a farm of servers (general purpose computers, file servers, and dedicated servers for mail, WWW, FTP, etc.) and a network of workstations and PC microcomputers. The departmental general purpose computers are an Alpha based Citum Power System (a repackaged Aspen server), a SPARCserver 670MP and a SPARCserver 10. (The SPARCservers will soon be replaced with a more powerful server.) The main file servers are Pentium based systems running Linux and utilizing RAID technology. The total disk space is currently well over 100 Gbytes. All the Alpha and Pentium based machines use Linux, but the SPARC computers run SunOS/Solaris. Together these systems support a wide variety of services, languages and software tools including electronic mail and news, graphics and visualization tools, several typesetting systems, and relational database systems. Special attention has been placed on security and reliability.

The workstation network consists of about 10 SPARCstations and about 280 PCs (mostly Pentiums with high resolution monitors) running Linux. Windows 3.1, Windows 95 or Windows NT can be used as an alternative for Linux. About 30 of the Linux workstations are mobile laptops which can join and leave the network dynamically. Networking is based on an ATM backbone with Ethernet edge devices. The department has six ATM-switches installed. The mobile laptops can also utilize a departmental 1 Mbit/s radio network which currently has 7 base stations. On the UNIX side (Linux, SunOS/Solaris), NFS is used to share common resources. On the Windows side, Samba (a UNIX hosted Lan Manager Server) is used. The workstations are used as tools for software develop-
ment, in research and all levels of teaching.

The network of the department is connected to the university backbone network, giving access to computers at the University Computing Centre as well as to the FUNET wide area network that links Finnish universities and research establishments. The computers operated by the Computing Centre include SPARC (Sun, Solbourne, Axil), Digital Alpha and HP machines running under UNIX. Services provided by the Computing Centre include the Oracle and Ingres database management systems, the SAS statistical analysis package, the NAG numerical library, and Pascal, Ada, and Prolog programming environments.

In addition, the department has access to a Cray C94, a Cray T3E, two SGI Power Challenge/Onyx, a Digital AlphaServer 8400, and other supercomputers at the Center for Scientific Computing.

The national FUNET network is further connected to the Nordic University Network, Nordunet, with a dedicated G.703(E3) 34 Mbit/s line. The Nordunet has also a 79 Mbit/s terrestrial connection to NAP Pennsauken in the United States as well as many connections to the European network infrastructure. This means that the department is very well connected to the Internet.
Chapter 7

International Relations

Student Exchange

The department is involved in the Sokrates/Erasmus programme (European Community Action Scheme for the Mobility of University Students). Through this programme students of the department have an opportunity to study at several European universities, including the Technical Universities of Darmstadt, Graz and Munich, the University of Bath, the University of Catalonia in Barcelona and the National Institute of Applied Sciences in Lyon.

The department also accepts visiting students from these and other partner institutions. Exchange of students within the Nordic countries is also easy with the financial support of NORDPLUS, a program coordinated by the Nordic Council of Ministers.

Postgraduate studies at the department can usually be accomplished in English. The department may admit foreign graduate and postgraduate students, who want to study for either the M.Sc. or Ph.D. degree, or visiting students. The Academy of Finland and the Ministry of Education and Science offer grants for foreign postgraduate students within the international exchange programs. Postgraduate studies are usually performed within one of the graduate schools, HeCSE or HeCSE-
http://www.cs.helsinki.fi/combi (see Section 4.2).

Research Cooperation

The department has active contacts with many European and American computer science departments. Faculty members have joint research efforts with individual researchers from abroad. The results of this international cooperation are partly presented in the publication list. Many fac-
ulty members have spent long periods of time abroad as visiting researchers or guest professors. In turn, apart from more than 20 short term visitors every year, the department has hosted several visiting foreign scholars. A list of long term visitors in 1996–98 is given below.

The department has actively taken part in the European research cooperation. The department has participated in the COST 247 project *Formal methods in Communication Protocol Design*.


In addition the department is a member institution of EC ACTS-project *DOLMEN Service Machine Development for an Open Long-Term Mobile and Fixed Network Environment* (AC036, Kimmo Raatikainen, 1995–98). The project *Object-oriented programming and compiler construction* (Jukka Paakki, 1995–97) was funded jointly by the Academy of Finland and Deutscher Akademischer Austauschdienst.

**Faculty Members Abroad in 1996–98**

Helena Ahonen, Eberhard-Karls-Universität Tübingen, Germany, 8/97–
Tapio Elomaa, Institute for Systems, Informatics and Safety, Joint Research Centre, European Commission, Ispra (Va), Italy 2/97–
Gösta Grahne, Université de Paris-Sud, Centre d’Orsay, France, 5/96
Gösta Grahne, Concordia University, Montreal, Canada, 9/97–
Kari Granö, Technische Universität München, Germany, 2–4/96
Teemu Kerola, University of Texas, Austin, USA, 8/97–7/98
Jyrki Kivinen, University of California at Santa Cruz, USA, 12/97
Heikki Mannila, Max Planck Institut für Informatik, Saarbrücken, Germany, 9/95–2/96
Jukka Paakki, Technische Universität München, Germany 2–3/96
Juha Taina, University of Virginia, Charlottesville, USA 9–12/96
Martti Tienari, University of British Columbia, Vancouver, Canada, 5-7/96
Martti Tienari, La Trobe University, Bundoora, Australia, 8/96
Olli Yli-Harja, University East Anglia, Norwich, UK, 1–5/98

Foreign Visitors in 1996–98

Prof. Amihood Amir, Bar-Ilan University, Israel, 10/97
Dr. Hiroki Arimura, Kyushu University, Fukuoka, Japan, 5–10/96
Dr. Jean-François Boulicaut, INSA, Lyon, France, 11/97-8/98
Dr. Alvis Brazma, University of Latvia, Riga, 8–12/96, 1/98
Dr. Peter Cheeseman, NASA Ames Research Center, Moffett Field, CA, USA, 5/97
Dr. Gautam Das, University of Memphis, USA, 8/96, 5–6/97, 5–6/98
Dr. Dimitri Dracopoulos, Brunel University, London, UK, 8/96
Prof. Jürgen Eickel, Technische Universität München, Germany, 10/97
Dr. Ronen Feldman, Bar-Ilan University, Israel, 8/96
Dr. Peter Grünwald, CWI, Amsterdam, The Netherlands, 5/97-7/97
Dr. Dimitrios Gunopulos, IBM, Almaden, USA, 9–10/96
Mr. Alberto de las Heras, Universitat Politecnica de Catalunya, Spain 3/97-7/97
M.Sc. Riitta Höllerer, Technische Universität München, Germany, 9/96, 9–10/97
Dr. Inge Jonassen, University of Bergen, Norway, 11/96, 1/98
Dr. Roni Khardon, Harvard University, USA, 10/96
Prof. Hartmut König, Brandenburgische Technische Universität Cottbus, Germany, 8/97
Prof. Gad M. Landau, University of Haifa, Israel, 10/97
B.Sc. David Nespoli, USA, 1/96–12/96, 6/97
Mr. Torsten Polle, Universität Hildesheim, Germany, 8/96
Prof. Thomas Richardson, University of Washington, Seattle, USA, 4–7/97
Mr. Jacques Scubic, INSA, Lyon, France, 6/97
Dr. Kuldar Taveter, Tallinn Technical University, Estonia, 10–11/96
Prof. Leo Vohandu, Tallinn Technical University, Estonia, 12/96
Dr. Son T. Vuong, University of British Columbia, Vancouver, Canada, 6/97
Prof. Emmanuel Waller, Université de Paris-Sud, France, 8–9/96
Mr. Hannes Wettig, Universität Köln, Germany, 8/96-7/97

Other Activities

Faculty members regularly serve as referees for international computer science journals and conferences (see Section 3). Some faculty members are also reviewers for various review publications, such as Computing Reviews and Mathematical Reviews. Many faculty members have served in the program committees of various conferences (see Section 3). Prof. Esko Ukkonen is the editor-in-chief of the Nordic Journal of Computing. Prof. Heikki Mannila is the editor-in-chief of the Journal of Data Mining and Knowledge Discovery. Assoc. Prof. Matti Mäkelä is an international coordinator of ACM SIGNUM Newsletter.

The department hosted the joint standardisation meeting of ISO/IEC JTC1/SC21 WG7 and ITU-T SG4 Q.12/Q.14 in Helsinki, 14–25 July, 1997. The joint meeting covered 12 separate standardisation issues, and about 60 delegates were present from 13 countries.

The department organized the 6th Scandinavian Conference on Artificial Intelligence in Helsinki in August 1997. About 80 researchers took part. The conference proceedings were edited by Gösta Grahne and published by IOS Press in the Frontiers in Artificial Intelligence and Applications Series.