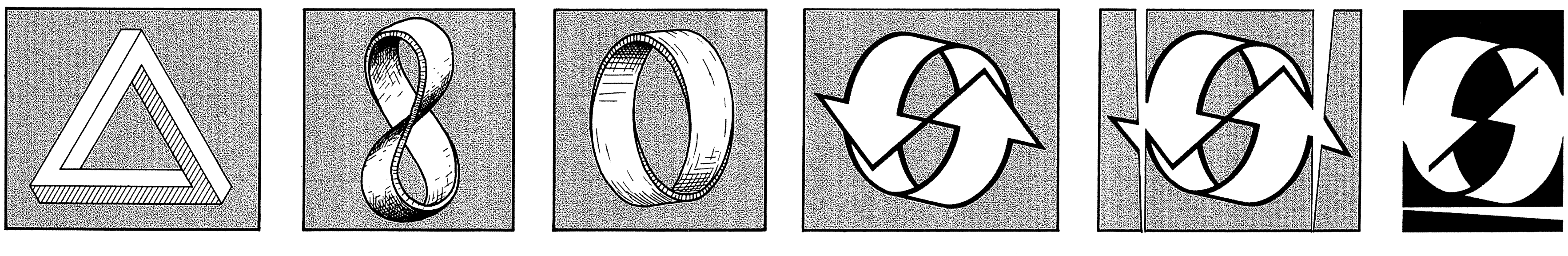
ANNUAL REPORT

CWI

Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science



ANNUAL REPORT 1989

Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

Kruislaan 413 1098 SJ Amsterdam
P.O.Box 4079 1009 AB Amsterdam the Netherlands

The Stichting Mathematisch Centrum was founded on February 11 1946, as a non-profit institution aiming at the promotion of mathematics, computer science, and their applications. It is sponsored by the Dutch Government through the Netherlands organization for scientific research (NWO).

Board of Directors

P.C. Baayen (scientific director)
J. Nuis (management director)

CWI
Centrum voor Wiskunde en Informatica
Centre for Mathematics and Computer Science

Kruislaan 413	P.O. Box 4079
1098 SJ Amsterdam	1009 AB Amsterdam
The Netherlands	The Netherlands
Telephone	020-592 9333
(from abroad)	31 - 20 592 9333
Telefax	020-5924199
(from abroad)	31 - 20 5924199

Contents

Financing
The years 1984-1988 saw strong growth at CWI. Chiefly thanks to the government backed Information Technology Promotion Plan (INSP) for this period, CWI was able to expand its computer science research and hence reinforce its position as a leading centre of research. Among the consequences of this expansion was the creation of a good basis for CWI participation in various national and international programmes like SPIN and ESPRIT. This potential was exploited to the full and led to further contacts. Hence, this reinforcement contributed (perhaps even decisively) to CWI becoming a full partner in a formal cooperative agreement with two far larger institutions, GMD of West Germany and INRIA of France (more on this later). Moreover, in April 1988, the report of an international committee of evaluation set up by The Netherlands Organization for Scientific Research (NWO) concluded that CWI had made excellent use of the Dfl. 2 million annual INSP grant. The committee went on to say that this support should be made structural—and at a higher level—when the INSP grant came to an end in 1989. Unfortunately the recommendation was only partially followed; the extra grant was to be mainly temporary and not for the full amount requested. Added to this, more promotional schemes ended in the same year, and the subsequent phase of European promotion of information technology increasingly focused on applications. For CWI, which plans to maintain its position as an institute for fundamental research, this development offers less potential for linkage. Income from commissions and joint projects with third parties was also lower than expected. All in all, it is clear that the second half of 1989 was a difficult time for CWI, and that some cutbacks were inevitable. In the meantime it has been decided to market CWI more actively in areas such as outside commissions, joint-projects with third parties, and course organization.

Evaluation
CWI's Policy document for the years 1988-1993 states that all research sectors will be regularly evaluated by aptly named international visiting committees. In 1987 a start was made with three areas: statistics, stochastics and system theory. In October 1989 it was the turn of five more specialties: algebra, analysis, geometry, optimization and numerical mathematics. The committee, under the chairmanship of R. Tijdeman (Leiden University), comprised D. Gorenstein (Rutgers University, New Brunswick), R.L. Graham (AT&T Bell Labs, Murray Hill), W. Hackbusch (Kiel University), J. Jäger (University of Heidelberg), J. Korevaar (Amsterdam University), G. Nemhauser (Georgia Institute of Technology, Atlanta), and M.N. Spijker (Leiden University). In general the evaluation was very favourable for CWI. Among the recommendations were: fewer and larger project groups; more attention to experimental ('computational') and applied mathematics; reinforcement of the synergy between mathematics and computer science; stronger links with industry; and expansion of consultancy services.

ERCIM
The cooperative agreement started in 1988 between CWI and the Gesellschaft für Mathematik und Datenverarbeitung (GMD) of West Germany, and France's Institut National de Recherche en Informatique et en Automatique (INRIA), acquired its first teeth during the year under review. The second joint workshop was held in April at Schloss Birlinghoven (GMD's headquarters); the themes were *Theoretical and pragmatic issues of programming, Concurrency* and *Object-oriented graphics and interfaces.* During the workshop the directors of the three institutes signed a formal agreement on joint activities. This was soon followed by publication of the first GMD-INRIA-CWI Newsletter; a second issue appeared in November. Activities under the cooperative agreement have been given the acronym ERCIM (European Research Consortium for Informatics and Mathematics). Possibilities for the immediate future include serious consideration of admission of additional, national institutes. The third workshop was held at INRIA's headquarters in Rocquencourt in December. The subjects dealt with were *Cryptography and security, VLSI design* and *Scientific computing.* The finishing touches were also put to the first advertisement for applicants for three post-doctoral fellowships established by ERCIM. In March a meeting was held at CWI to brief the West German and French technical/

scientific attachés on the cooperation with GMD and INRIA.

Projects

INSP support helped CWI find a role in European research programmes - ESPRIT first and foremost. The second phase of this project, which started in 1989, is more application oriented than its predecessor. A separate and relatively smaller ESPRIT BRA (Basic Research Action) programme was also started up. CWI succeeded in finding a place in both programmes, with three projects (ESPRIT II: *Atmosphere, GIPE II* and *TROPICS*; ESPRIT BRA: *Concur, Integration* and *Semagraph*).
CWI also participates in RACE (R&D in Advanced Communications-technologies in Europe) with two projects, *RIPE* and *SPECS*, and in BRITE-EURAM (Basic Research in Industrial Technology for Europe - European Research in Advanced Materials) with research into adaptive multigrid methods for solution of Navier-Stokes equations, and in the European programmes *Ada* and *Euromath.* For more details see the list of (inter)national programmes in which CWI participates, later in this Annual Report.

In fall 1989, CWI started research into natural languages in relation to expert systems as part of its *Logical aspects of artificial intelligence* project.

Selected key points from the remaining 30 odd ongoing projects are detailed below.

CWI is engaged in research in the framework of some Technical Sciences Research Foundation (STW) projects. From the outset these projects are attuned with the needs of industry or government. In the *New architecture for interactive raster graphics based on VLSI,* a chip has been developed which enables faster (by a factor of 100) generation of raster pictures. This brings within reach the use of realistic pictures within an interactive environment. Given market interest in this chip, CWI will continue development beyond the prototyping phase usual in scientific circles. Further, 1989 saw the completion of two projects in the area of traffic: *Prediction and control problems for the Dutch freeway control and signalling system,* and *Overload control for communication systems.* CWI worked jointly with the University of Twente on all of these STW projects, and also with Philips Telecommunication on the third.

The international Euromath project (1987-1992) reached the end of its first phase in mid-1989. The project aims to give European mathematicians a fully-customized, integrated research environment based on modern information technology. Planned facilities include an electronic conferencing facility, production and exchange of mathematical publications, and data banks with information on articles, mathematicians, conferences, etc. CWI carried the chief responsibility for functional design of the Euromath system. This design - and a pilot implementation to test ideas - was completed in 1989. The second phase - involving definitive implementation of the system - is expected to start during 1990.

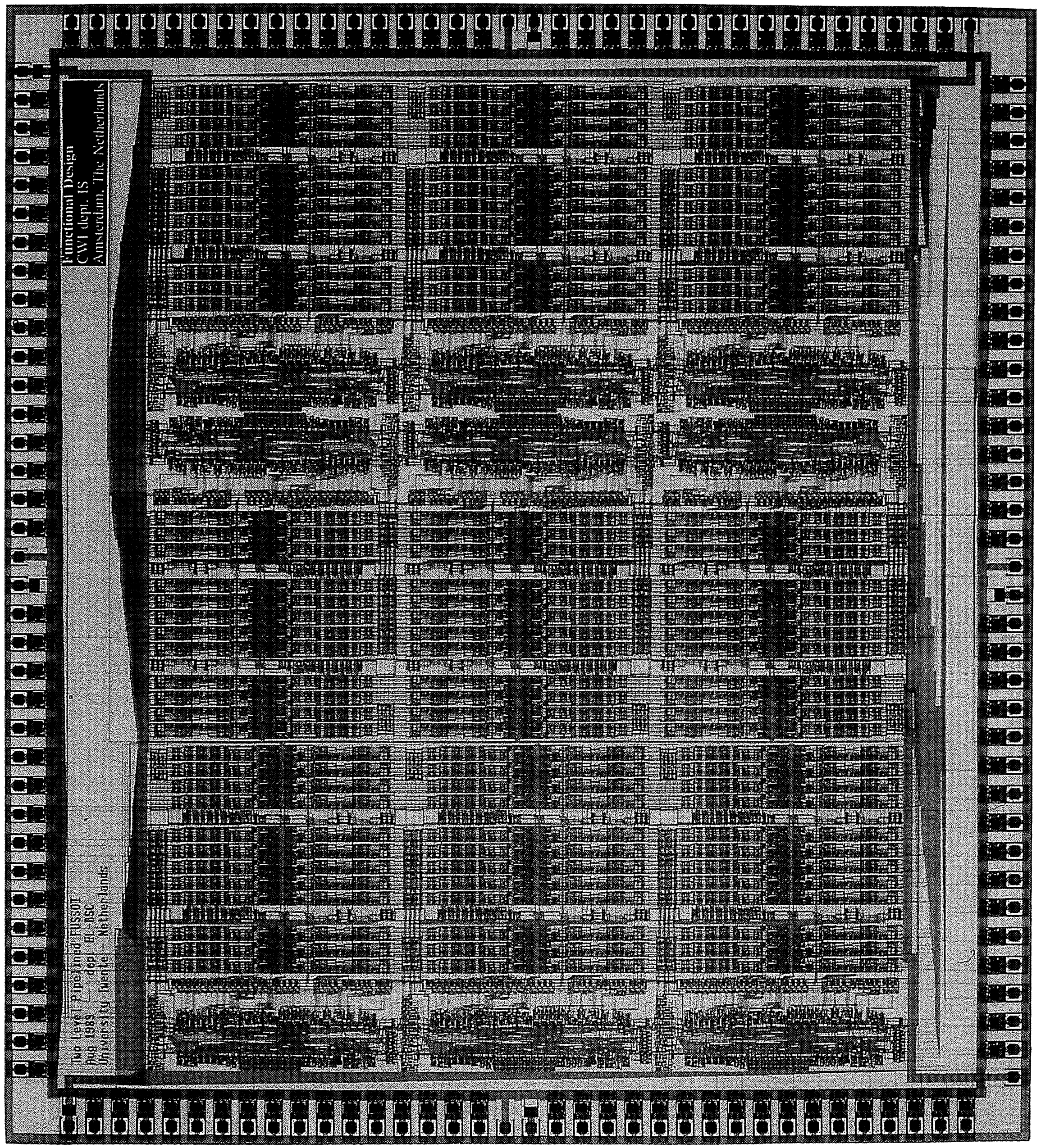
CWI's contract research as part of the ESA (European Space Agency) Hermes project was extended to 1990. This research concerns *Efficient techniques for solution of the Euler and compressible Naves-Stokes equations.* 1989 saw development of a damping technique for nonlinear multigrid, as well as a direction-dependent prolongation technique. Both were successfully applied to the computation of a hypersonic reentry flow around a blunt-nosed aircraft shape.

Research within the framework of the National Innovative Research Programmes (IOP) concerned *Reliable and efficient methods for semi-conductor equations,* in close cooperation with Philips CFT Centre in Eindhoven; an adaptive multigrid system has already been developed. The project runs until 1992.

Lastly, the *Amoeba* project (a distributed operating system jointly developed at CWI and the Free University of Amsterdam), received support from the Open Software Foundation (OSF). This support is intended to aid adequate distribution of the systems. The CWI research is part-sponsored by the DEC company's European External Research Programme.

Conferences and courses

CWI has always been active in (joint-) organization of conferences, courses, workshops, *This chip shown is the last step of a fast raster picture generator. Object descriptions are decomposed in colour functions along a scanline at a constant rate of 60Hz. A large array of processing elements (the chip shown contains 9 of these elements) generates from these colour functions a sorted stream of colour values which can be directly converted in an analogue video signal. The chip was developed by CWI in cooperation with the University of Twente.*



colloquia and suchlike. This type of activity - a vital element for the institute in its centre-role - increased again in 1989, and will receive even more attention in the future. Together with the Free University of Amsterdam, CWI organized the 8th biannual International Symposium on *Mathematical Theory of Networks and Systems* (MTNS-89); there were 300 participants. CWI was also involved in the organization of VLDB '89, the 15th International Conference on *Very Large Data Bases* which brought 400 participants to Amsterdam in August. On a somewhat smaller scale CWI organized international workshops on *Intelligent CAD systems* and *Stereology & Spatial Statistics, Stochastic Mathematics and Image Analysis* (35 and 55 participants). A series of colloquia started up in autumn 1989, in cooperation with the 'History and Social Function of Mathematics' national working party; the theme was *The history of (electronic) computing.* Among the many courses organized was the traditional vacation course for school teachers (this time on *Mathematics in Holland's 'golden' - 17th - century*). A non-specialized summer course on Fractals drew some 30 participants; specialized courses covered *Parallel computation* and *Networks.*

Miscellaneous

There were several additional noteworthy events and circumstances during the year under review.

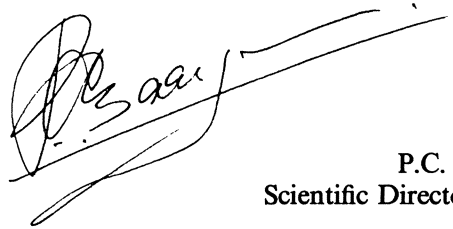
The expertise centre CAN (Computer Algebra Nederland) started up in September, with CWI acting as host. The initial government grant of Dfl. 1.7 million covers facilities and services (including consultancy) for computer algebra systems researchers and users over a period of three years. A symposium with some 100 participants marked the official opening in December.

For seven years CWI has acted as one of the central nodes in the European network of UNIX-users, and as 'gateway' between Europe and the USA. In 1989 the European UNIX Users Group replaced CWI's existing VAX equipment with a SUN-4/280. This now serves around 1,500 European institutions.

In autumn 1986 a three-year general agreement was signed between IBM and the two Amsterdam universities plus CWI. This covers some tens of projects in the area of computer driven education and research, particularly in the alfa and gamma sciences, with IBM providing equipment and manpower back-up. CWI involvement was relatively modest and covered three projects. In October the agreement was extended for a further three years.

There were three 25-year jubilees in the year under review. Two of these concerned departmental heads—P.J. van der Houwen (Numerical Mathematics) and J.W. de Bakker (Software Technology); the third was Management Director J. Nuis. Each event was celebrated with a symposium: *Construction of stable numerical methods for differential and integral equations; J.W. de Bakker, 25 years of semantics;* and *Computational Engines* (on the pre- and early history of the electronic computer).

As shown in this introduction, CWI continued developing a broad range of activities during the year under review. The more stringent financial situation has in no way dulled our determination to achieve the aims stated in the Policy Document 1988-1993. I am convinced that the extra effort CWI has invested on a wide front will pay dividends in the future.



P.C. Baayen
Scientific Director CWI

Organization

The Centre for Mathematics and Computer Science (CWI) is the research institute of the Stichting Mathematisch Centrum (SMC), which was founded on 11th February 1946. SMC falls under The Netherlands organization for scientific research (NWO), the main source of funding.
In line with its statutory purpose 'to foster the systematic pursuit of pure and applied mathematics and computer science in The Netherlands', SMC immediately set up an institute for fundamental research, the Mathematical Centre. From the outset this institute played an important role in the development of computer science in The Netherlands. A change to the present name, CWI, in September 1983, reflected the major expansion of research in this field. On the national level this growth led to the setting-up in 1982 of the Stichting Informatica Onderzoek in Nederland (SION), an independent NWO research organization for computer science. Its formal connection with SMC is embodied in a Permanent Consultation Commission.

SMC also finances research projects at Dutch universities. These projects are organized in eight national working parties in the following fields:

- Numerical mathematics;
- Stochastics;
- Discrete mathematics;
- Operations research and system theory;
- Analysis;
- Algebra and geometry;
- Logic and foundations of mathematics;
- Mathematical physics.

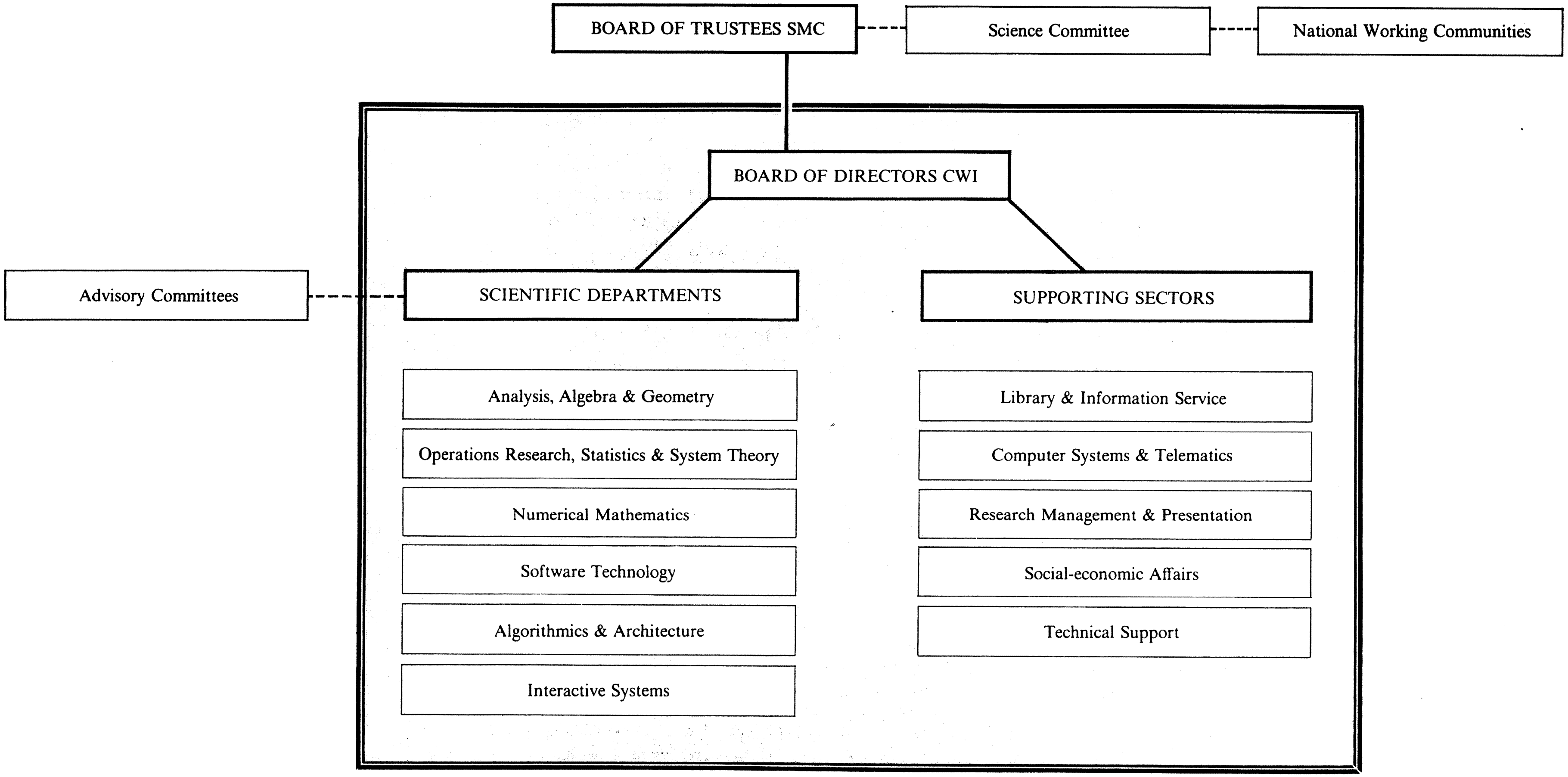
SMC also supports the national working party on History and Social Function of Mathematics.

SMC is administered by a Board of Trustees. Actual administration is delegated to the Board of Directors of SMC, which is also responsible for CWI. A Science Committee advises the Board of Trustees on matters of research policy and organization involving both the National Working Parties and CWI. The Science Committee is made up of researchers from universities and CWI. A number of Advisory Committees make recommendations to CWI scientific departments on implementing research plans.
Research at CWI is also evaluated by international visiting committees. The first evaluation, in 1987, covered statistics, stochastics and system theory. The next, in 1989, dealt with algebra, analysis, geometry, optimization and numerical mathematics.

CWI's goal is fundamental and advanced research into mathematics and computer science, with special emphasis on areas to which the research may have relevant applications. Research is fundamental in that it mainly concerns those problems lacking standard methods of solution. It is advanced, in that CWI aims at a high level, both nationally and internationally. Preference is given to subjects with internationally relevant development potential.

The organization structure of SMC and CWI is shown on the opposite page. In this scheme, the department of Computer Systems & Telematics comes under supporting sectors. However, because of the increasing emphasis on research, its activities are represented as those of a scientific department. The departmental structure is less rigid than it appears, given considerable inter-departmental collaboration. This is a matter of deliberate policy, not only in the selection of research topics, but also in the selection of the permanent scientific staff. There are also a number of supporting service departments.

By international standards CWI might appear relatively small and incapable of involvement in the full range of major developments in mathematics and computer science. However, size can be deceptive. By its very nature CWI, with its close knit research units supported by state-of-the-art computer facilities and a well stocked library, is ideally equipped to handle the dynamic and interdisciplinary demands of present day research.



Organizational chart: the Stichting Mathematisch Centrum SMC and its research institute, the Centrum voor Wiskunde en Informatica (Centre for Mathematics and Computer Science) CWI.

M. Hazewinkel (head of department)

G. Alberts	P. Hofstee	H.N.M. Roozen
J.T.M. van Bon	H. Inaba	S.N.M. Ruijsenaars
A.E. Brouwer	Jin Cheng-fu	G.C.M. Ruitenburg
A.M. Cohen	T.H. Koornwinder	H.A.J. Schellinx
O. Diekmann	M. Kretzschmar	J.K. Scholma
M. Dijkhuizen	J. van de Lune	C.P. Schut
G. Greiner	J.A.J. Metz (advisor)	B. de Smit
F.C.A. Groen (advisor)	J. van Neerven	N.M. Temme
J.A.P. Heesterbeek	A.B. Olde Daalhuis	J. de Vries
H.J.A.M. Heijmans	J.B.T.M. Roerdink	M. Zwaan

INFECTIOUS DISEASES IN STRUCTURED POPULATIONS

Introduction
Epidemiology is the scientific description of the distribution in space and time of diseases and the search for factors responsible for the observed patterns of distribution. Originally most emphasis was put on the study of infectious diseases, but in the second half of this century and in the developed world the study of degenerative diseases such as cancer and cardiovascular diseases became predominant. The current AIDS epidemic has led to an upsurge in interest for infectious diseases.

CWI's joint project with the University of Leiden focuses on populations where heterogeneity among individual characteristics can strongly influence the spread of an infection. The aim of the project is to provide an overall picture of the common mathematical structure of epidemic models or, more generally, to provide a survey of the various structures inherent in such models, and their interrelations.

Mathematical questions in epidemiology
The understandable expectation that mathematical models are used for prediction of future trends in the spread of infectious diseases is unwarranted. Mathematical models are used for obtaining insight, in particular concerning the relative importance of factors influencing the spread of the infection and, more generally, concerning the relation between mechanisms on the individual level and phenomena at the population level.
One of the major questions that is approached from a mathematical angle is the *invasion problem*: can an infection, when introduced in a susceptible population, cause a spreading epidemic - or will it disappear? A threshold-quantity called the *basic reproduction ratio* can be used to answer this question and its existence is a major insight that mathematical thinking has brought to epidemiology (more about this in the next sections). The basic reproduction ratio is also important in studying the efficacy of different control measures and as a tool for discriminating between different vaccination strategies.

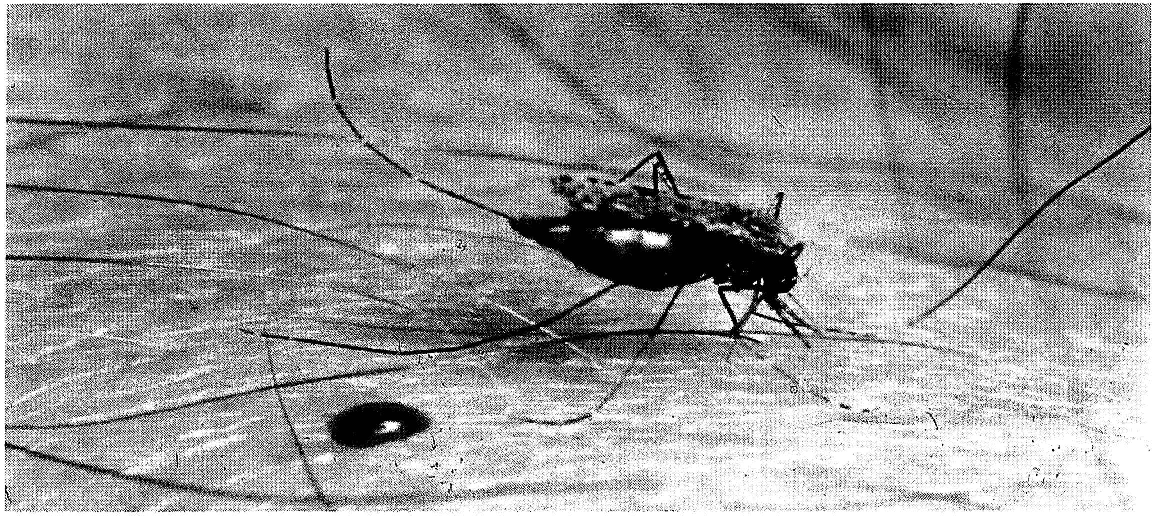
Once an epidemic occurs, the question is how the fraction of infected individuals will develop in time and what fraction ultimately escapes from getting the disease. A second major insight furnished by mathematical modelling is that there will always be a positive fraction of individuals that do not contract the infection. For the questions mentioned so far one assumes that the time scale of the epidemic process is much faster than the time scale of demographic changes in the population. If an infection has established itself in a population it is said to be endemic. Now we look on longer time scales and the demography becomes important. Questions concern the stability of the endemic state, the possibility of oscillations, and the problem of *regulation*: how does the disease affect population growth?

Two successful examples

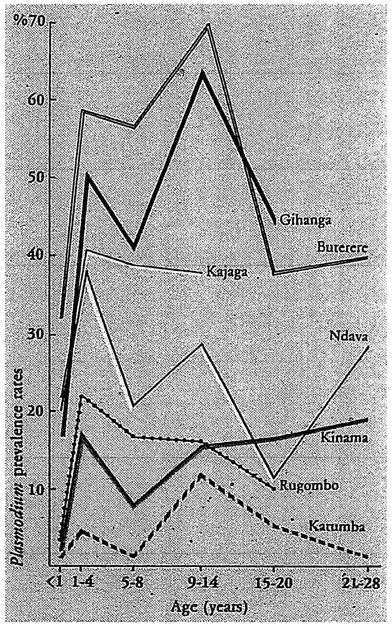
In studying these questions modelers have had an impact on public health decisions. Let us discuss two of these success stories of mathematical models in epidemiology.

The first is probably the earliest example of insightful application of mathematics to epidemiology. It dates back to 1909 when Ross introduced the notion of a threshold quantity already referred to above. He observed that eradication of malaria is possible by decreasing the number of mosquitos present in a certain area. Prior to that it was generally believed that malaria would always survive as long as some mosquitos were still present and that total eradication of mosquitos was impossible. Using a simple model, Ross showed that there was a quantity which, when suppressed below unity, would guarantee the disappearance of malaria from the area, and that this quantity depended on the ratio of mosquito density to human density. Thus he found a critical mosquito density. Empirical corroboration was later obtained in India with the discovery of neighbouring areas with and without malaria and mosquito densities respectively above and below the critical level.

A second example is the evaluation of different vaccination strategies. For example for rubella (German measles), which is a mild illness in most individuals but is a serious threat to pregnant women and their unborn offspring, there are three strategies. One is to vaccinate all young children, a second is to

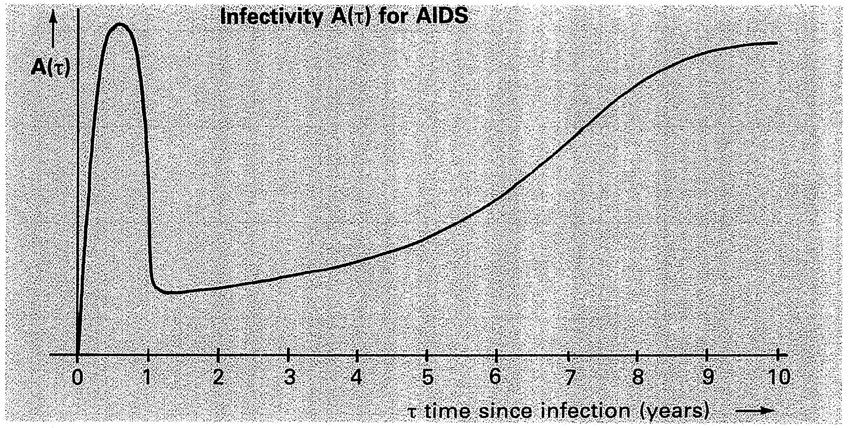


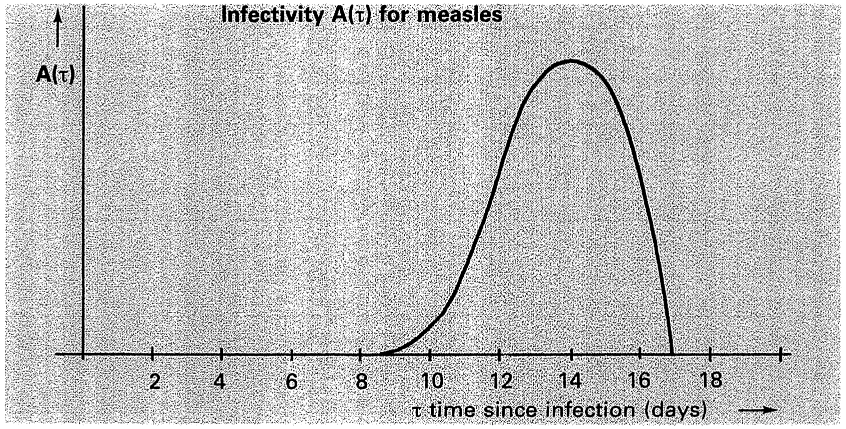
△
Anopheles stephensi, *female specimen (males don't bite), takes a meal of blood and gorges until overflow (drop on rear). This is how the malaria parasite is conveyed from mosquito to man and vice versa.*



◁*Malaria is caused by the parasite of the genus* Plasmodium. *Here we see the age-specific* Plasmodium *prevalence rates, i.e. the fraction of the population carrying the parasite, in several villages on the Ruzizi plain (Burundi). Malaria prevalence varies widely between villages, thus corroborating the idea of a critical mosquito density below which malaria disappears.*

For both pictures: courtesy H.J. van der Kaay, University of Leiden, Laboratory for Parasitology.





The infectivity A as a function of the time τ since infection for the diseases HIV (which causes AIDS) and measles. The characteristic function A can differ widely for various diseases. Note for example the difference in time scales and in the recurrency pattern.

vaccinate only prepubertal girls, and a third is a kind of combination: vaccination of all children at young age (about 1 year old) and again at about 11 years old. The second strategy was used in The Netherlands up to 1984, when a switch was made to the third. Mathematical analysis shows that certain strategies can in fact increase the fraction of serious cases. This is because complications of rubella infection are more likely to occur at a higher age, and that the average age at which infection is contracted can rise when using certain strategies. Important questions which can be answered by mathematical analysis, e.g. by studying the effect of vaccination on the threshold quantity, are:

- which immunization coverage is required to eliminate rubella? (For The Netherlands the answer is: around 95% of young children, a percentage attainable only in highly developed countries.)
- if elimination cannot be attained, what consequences does this have for the fraction of unvaccinated individuals?

It is thought that with The Netherlands' current combined vaccination strategy against measles, mumps and rubella it should be possible to eliminate these three diseases in the first half of the nineties.

The basic reproduction ratio

A paper by W.O. Kermack and A.G. McKendrick which appeared in 1927 in the Proceedings of the Royal Society at Edinburgh has had a major influence on mathematical modelling of epidemics. Nowadays most people refer to a certain simple system of ordinary differential equations as *the* Kermack-McKendrick model, whereas in fact they treated a much more sophisticated model. Their key idea was to describe the *average infectivity of an individual τ units of time after it became infected* by a non-negative function $A(\tau)$. The assumption is that infection triggers an autonomous process which develops within the host without any further influence of the environment and that, consequently, we can use an 'age' representation to describe the average infectivity. Aspects as the detailed stochastic modelling of the time evolution of the internal population of viral particles or bacteria and the concomitant reaction occurring in the immune system are incorporated in the function A; and in considerations where we need not specify A we do not have to worry about these.

Assuming that the number of contacts between susceptibles and infectives is proportional to the density of susceptibles times the density of infectives (the law of mass action), Kermack and McKendrick were led to introduce $R_0=S\int_0^\infty A(\tau)d\tau$ as the expected number of secondary cases (new infected individuals) produced by one infectious individual during its entire infective life in a susceptible population of density S. Clearly the R_0 has *threshold value* one, i.e. when $R_0<1$ no epidemic develops upon introduction of the infection in the population, whereas when $R_0>1$ an epidemic gets started. R_0 is called the *basic reproduction ratio.*

Heterogeneity

Things become more complicated when not all individuals are equally susceptible. Disease transmission will reflect differences in susceptibility and we have to carry out the right averaging procedure to arrive at R_0. The bookkeeping should take into account the different structures and distribution of the population with respect to these. In recent years considerable attention has been paid to the modelling of such more complex - and realistic - systems. The project presently described reflects this interest.

We introduce a variable ξ which we call the h-state (h for heterogeneity). ξ can be static or dynamic, discrete or continuous. Examples are age, sex, sexual activity, spatial position. The h-state space, i.e. the range of all relevant ξ, we call Ω. Let $A(\tau,\xi,\eta)$ be the expected infectivity of an individual which was infected τ units of time ago while having h-state η, with respect to a susceptible of h-state ξ. So all medical, behavioural, physiological and social aspects which are relevant for disease transmission are summarized by A and we shall need sub-models later on to specify A on the basis of certain assumptions. Because in the parametrization only the h-state at infection time enters, it is not necessary - for the time being - to specify the dynamics of the h-state. Let $S(\xi)$ be the susceptible population density as distributed over Ω. The *next generation operator*

$$(K(S)\phi)(\xi)=S(\xi)\int_\Omega \int_0^\infty A(\tau,\xi,\eta)d\tau\ \phi(\eta)d\eta$$

tells us both how many secondary cases arise, and how they are distributed over Ω, when we start with a ‘distributed’ individual ϕ. We consider $K(S)$ as an operator on $L_1(\Omega)$. Note that $K(S)$ is a positive operator (since S and A are non-negative, $K(S)$ preserves non-negativity). So, as a rule, there is a dominant eigenvalue λ_d and $K(S)^n\phi\sim c(\phi)\lambda_d^n\phi_d$ as $n\to\infty$ for any non-negative ϕ, where ϕ_d is the eigenfunction corresponding to λ_d, and $c(\phi)$ is a constant depending on ϕ. In other words after transients, which reflect how exactly the epidemic started, have died away (i.e. for sufficiently large n) the next generation will be a factor λ_d larger than the current one and the distribution of new cases with respect to h-state will be invariant. Clearly the biological quantity we are after is mathematically described by λ_d or, in symbols, $R_0=\lambda_d$.

Under various special assumptions it is now possible to calculate the basic reproduction ratio when arbitrary heterogeneity characteristics which influence the spread of an infection are taken into account.

Colloquium on models for infectious diseases

The work on the basic reproduction ratio, described in the above section, came out of the preparations for a colloquium on *models for the spread of infectious diseases in structured populations* which was organized over the whole of 1989. A total of 35 lectures were given by (mainly Dutch) researchers from various disciplines; the audience comprised around 30 theoretical biologists, epidemiologists, and mathematicians. The work on vaccination strategies in The Netherlands mentioned above was presented by J.A.M. van Druten (Department of Medical Statistics, University of Nijmegen). The twelve excursions to the mathematical side of the subject by the organizers - O. Diekmann, J.A.P. Heesterbeek, M. Kretzschmar and J.A.J. Metz - which touched, among other things, on all questions mentioned in the second section, will be expanded into a book during 1990 and 1991.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Algebra, discrete mathematical structures and computer algebra

The project concerns the relation between graphs/geometries on the one hand and group theory on the other hand. The main interest lies in graphs and groups of Lie type. These studies have led to the development of algorithms and, later software packages, thus focussing in on computer algebra. (1981)

A.M. Cohen, J.T.M. van Bon, A.E. Brouwer, G.C.M. Ruitenburg, B. de Smit.

Univ. Nijmegen, Univ. Rotterdam, Free Univ. Brussels, Univ. Michigan, Univ. Eindhoven, Cal. Tech. Pasadena, UC Santa Cruz, Free Univ. Amsterdam, Imperial College, Univ. Cambridge, Inst. of System Studies Moscow.

Analysis and mathematical physics

This project involves (aspects of) harmonic analysis on homogeneous spaces, special functions, quantum groups, q-special functions, integrable dynamical systems, and topological dynamics and some of the manifold interrelations between these topics. There are four subprojects.

- Analysis on semisimple Lie groups and symmetric spaces and the connection with special functions. (1972)
  T.H. Koornwinder, M. Dijkhuizen.

  Univ. Leiden, Univ. Nijmegen, Univ. Delft, Univ. Groningen, Univ. Amsterdam, Univ. Leuven, Univ. Wisconsin, Univ. London, Japan, Univ. Tunis.
- Lie Algebras, Hopf algebras and integrable dynamical systems. (1982)
  M. Hazewinkel.

  Univ. Amsterdam, Univ. Twente, UCLA, Univ. Utrecht, MGU Moscow, LOMI Leningrad.
- Relativistic and quantum integrable systems. (1986)
  S.N.M. Ruijsenaars.

  Univ. Amsterdam, Australia.
- Dynamical systems. (1976)
  J. de Vries.

  Univ. Delft, Univ. Maryland, Univ. Tel-Aviv.

Nonlinear analysis and biomathematics

Analysis of ordinary, partial and functional differential equations and integral equations which correspond to mathematical descriptions of biological processes. Development of a general mathematical modelling methodology, in particular for the dynamics of structured populations. (1975)

O. Diekmann, G. Greiner, J.A.P. Heesterbeek, H.J.A.M. Heijmans, H. Inaba, J.A.J. Metz, J. van Neerven.

Univ. Leiden, Univ. Delft, Univ. of Tech. Helsinki, Free Univ. Amsterdam, Univ. Calgary, Univ. of Strathclyde Glasgow, Univ. Tübingen, Kyoto Sangyo Univ., Univ. Hiroshima, Univ. Arizona, Georgia Inst. of Technology, Univ. München.

Asymptotics

This project includes research on asymptotic expansions of integrals and differential equations, and solving problems on analysis and asymptotics (with numerical aspects) from physics, biology, and statistics. (1975)

N.M. Temme, A.B. Olde Daalhuis.

Univ. Groningen, Univ. Winnipeg, Univ. Knoxville, Univ. Maryland.

Image processing and reconstruction

- Research on mathematical aspects of image processing and reconstruction by means of mathematical and numerical analysis,

mathematical statistics and computer science;
- Development of algorithms and software;
- Contact with medical investigators, biologists and physicists, as well as with laboratories.

(1985)

J.B.T.M. Roerdink, F.C.A. Groen, H.J.A.M. Heijmans, P. Hofstee, M. Zwaan.

Philips Medical Systems Best, Univ. Nijmegen, Univ. Delft, IZF-TNO Soesterberg, Philips Research Lab. Brussels, Harvard Univ., Ecole Normale Supérieure des Mines Paris.

Miscellaneous

- Classical analysis and number theory. (1972)
  This project concerns the study of problems of a (numerical/analytical) number-theoretic nature.
  J. van de Lune.

  Bell Labs, Free Univ. Amsterdam.
- History of mathematization.
  History of mathematical activity in the Netherlands over the period 1945-1960, in particular the history of founding the Mathematical Centre and of setting up the study course of Mathematical Engineer. Both events are being considered on the one hand in the general history context of their time, on the other hand against the background of the preceding development in the relation between mathematics and application. (1988)
  G. Alberts, M. Hazewinkel.

  Univ. Twente, Univ. Amsterdam.

O.J. Boxma (head of department)
J.M. Anthonisse
A.J. Baddeley
D.M. Bakker
H.C.P. Berbee
J. van den Berg
J.L. van den Berg
J.W. Cohen (advisor)
A.L.M. Dekkers
F.A. van der Duyn Schouten
K.O. Dzhaparidze
S.A. van de Geer
A.M.H. Gerards
R.D. Gill (advisor)
W.P. Groenendijk
L.F.M. de Haan (advisor)
R. Helmers
J.A. Hoogeveen
M. Kuijper
B.J.B.M. Lageweg
J.K. Lenstra
M.N.M. van Lieshout
R.A. Moyeed
H. Oosterhout
M.C.J. van Pul
M.W.P. Savelsbergh
A. Schrijver
J.M. Schumacher
J.H. van Schuppen
A.W. van der Vaart
S.L. van de Velde
B. Veltman
P.R. de Waal
P. Wartenhorst
J.W. van der Woude

PERFORMANCE ANALYSIS OF POLLING SYSTEMS

Introduction

Queueing theory is concerned with the mathematical research of the performance of a system offering services for collective use. Such a system may be a hospital or bank, but it can also be a 'flexible manufacturing system', telephone exchange or computer network. The object of study is formulated in abstract terms as a network of service units and customers requiring services at those units. The nature of the arrival processes and service requests is usually such that they have to be represented by stochastic processes. Hence the most important performance measures, like waiting times, workloads and queue lengths, are random variables. Accordingly, the main techniques of queueing theory stem from probability theory.

In the beginning of this century, queueing theory was developed as a tool for dimensioning telephone exchanges. In the sixties and seventies it turned out that queueing models could also lead to accurate predictions of the behaviour of complex computer systems. This gave a strong impulse to the research on queueing networks. Today, the distributed nature of modern computer-communication networks poses a new challenge to queueing theory. We are beginning to see systems with distributed communications, distributed storage, distributed processing and distributed control. Unfortunately, a thorough understanding of the basic principles of distributed systems is still lacking. Such principles are needed to predict performance, to explain behaviour and to establish design methodologies.

At CWI one project is mainly devoted to queueing theory and its application to the performance analysis of computer and communication networks. Computer performance research carried out under the Government's Information Technology Promotion Plan INSP (1984-1989) has resulted in two Ph.D theses, to be defended in early 1990. The following exposition concerns the research area of W.P. Groenendijk's thesis, viz., distributed control of multi-access communication channels. Below we discuss the queueing model under consideration, and the main results obtained. But first, we describe the kind of computer-communication network that has stimulated this research.

Many communication systems provide a broadcast channel which is shared by all connected stations. When two or more stations wish to transmit simultaneously, a conflict arises. The rules for resolving such conflicts are referred to as 'multi-access protocols'. The token ring protocol is one such protocol used in many local area networks. In a token ring local area network, several stations (terminals, file servers, hosts, gateways, etc.) are connected to a common transmission medium in a ring topology [Figure 1]. A special bit

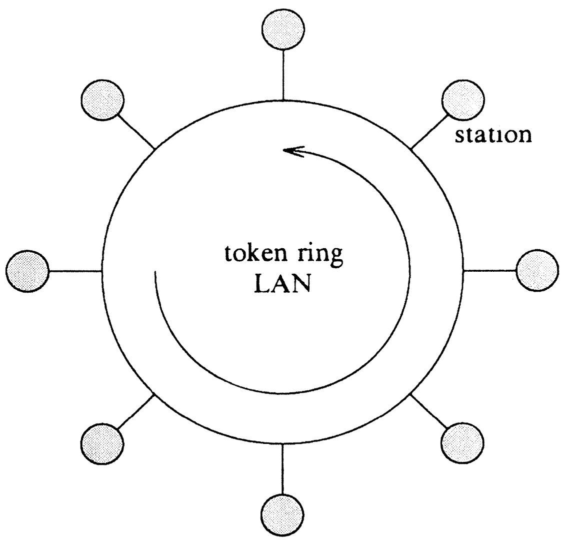


Figure 1. A token ring local area network.

sequence (called the *token*) is passed from one station to the next; a station 'possessing the token' is allowed to transmit messages. After completion of its transmission the station releases the token, giving the next station in turn an opportunity to transmit. This situation can be represented by the following queueing model - known as a *polling model.*

The polling model

A polling model is a single-server multi-queue model, in which the server attends to the queues in cyclic order [Figure 2]. The N queues $Q_1,...,Q_N$ have infinitely large waiting rooms. Arrival times of customers at the queues are usually assumed to occur according to a Poisson process. Service requirements of customers at a queue are independent, identically distributed stochastic variables; the same holds for the switch-over times of the server between queues. Arrival rates, service time and switch-over time distributions may differ from queue to queue.

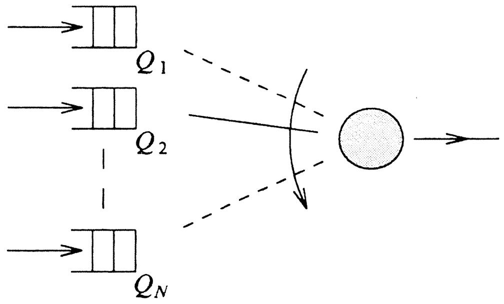


Figure 2. Queueing model of a polling system.

A polling model describes the behaviour of a token ring local area network in a natural way. The server represents the token-passing mechanism, and the customers represent messages generated at the stations. But many other situations in which several users compete for access to a common resource can be described by this polling model. Examples are a repair man patrolling a number of machines which may be subject to breakdown, assembly work on a carousel in a production system, and a signalized road traffic intersection. Depending on the application, various service disciplines at the queues may be considered. Common disciplines are *exhaustive service* (the server continues to work at a queue until it becomes empty), *gated service* (the server serves exactly those customers who were present when he arrived at the queue) and *1-limited service* (the server serves just one customer - if anyone is present - before moving on to the next queue).

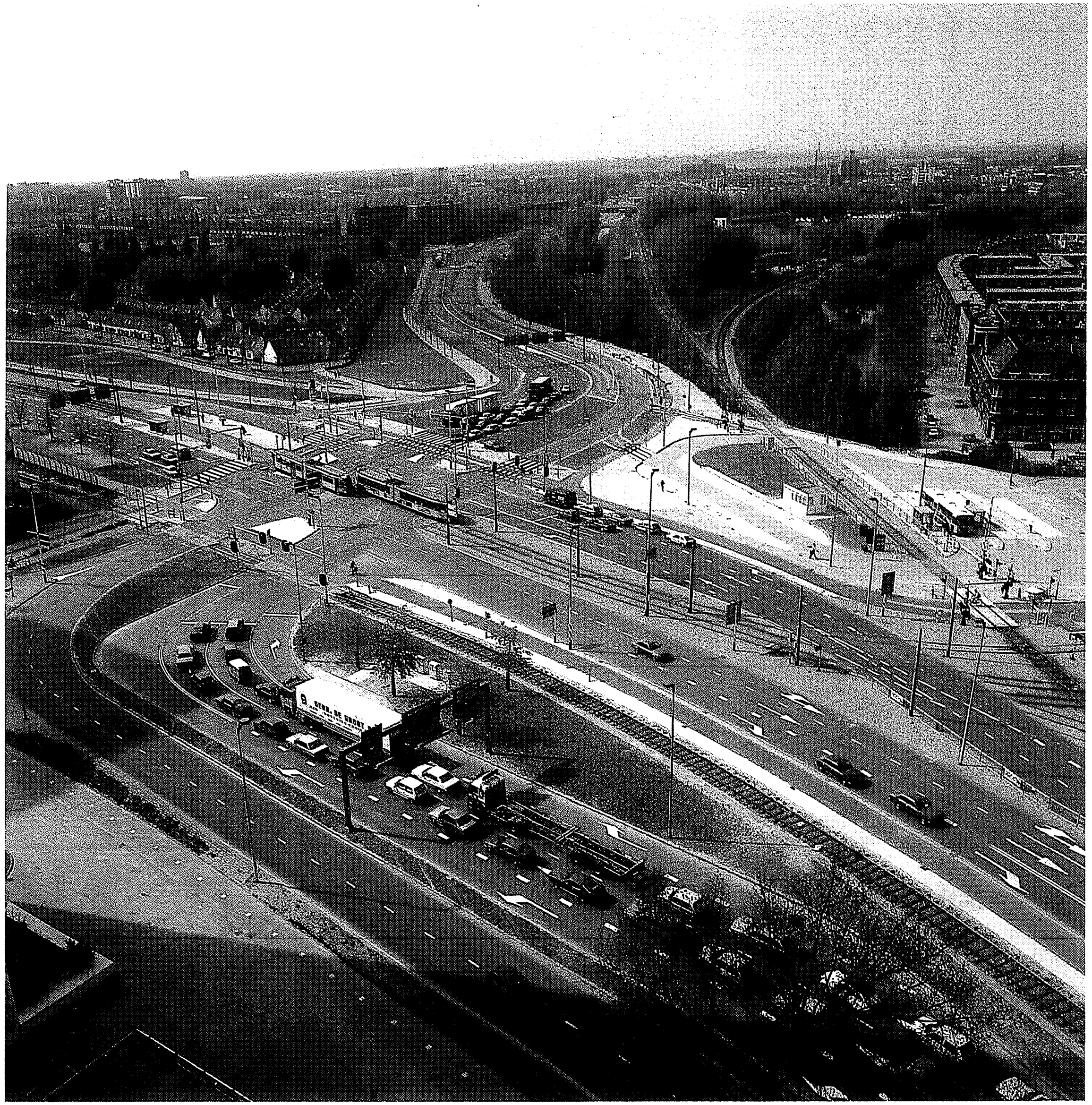
Analysis of the polling model

For polling models with exhaustive and gated service, the steady-state mean waiting times at all N queues can be determined by solving a set of $O(N^2)$ linear equations. The case of 1-limited service is much less amenable to an exact analysis. For only two queues, the joint queue length distribution at both queues can be determined by formulating and solving a boundary value problem for analytic functions, a Riemann or Riemann-Hilbert problem; the mean waiting times are thus also found - these being expressed as singular integrals. But, when there are at least three queues with 1-limited service, the mean waiting times are unknown. However, even in such a case there exists a simple expression for a certain weighted sum of all the steady-state mean waiting times. Twenty-five years ago Kleinrock showed that, under quite weak assumptions, in the case that all switch-over times between queues are zero,

$$\sum_{i=1}^{N} \rho_i E\mathbf{W}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)}. \qquad (1)$$

Queues are, of course, a very familiar phenomenon in traffic. A crossing with traffic lights is an example of a system of queues waiting to be served. Such a system bears similarities to situations inside computer and communication systems and may be modelled by using the same underlying queueing theory.

Photo: courtesy Nederland Haarlem B.V.



Here $E\mathbf{W}_i$ denotes the mean waiting time at Q_i, λ_i the arrival rate, ρ_i the offered traffic load and $\beta_i^{(2)}$ the second moment of the service requests; $\rho = \Sigma_i \rho_i$ denotes the total offered load. This is called a *conservation law*: if the service discipline at a queue is changed, the weighted sum of mean waiting times - i.e. the lefthand side of (1) - remains the same, although the individual mean waiting times may change.

Work conservation and work decomposition

The conservation law is a consequence of the 'principle of work conservation'. Suppose the scheduling policy, i.e., the procedure for deciding which customer(s) should be in service at any time, does not allow the server to be idle when at least one customer is present, nor does it affect the amount of service given to a customer or the arrival time of any customer. Comparing the sample paths of the 'workload process' for such a system under different scheduling disciplines leads to the observation that *the workload process is independent of the scheduling discipline.*

The principle of work conservation has proven very useful in the past. It enables one to analyze the workload process of queueing systems with a highly complicated scheduling discipline as if the scheduling discipline were a relatively simple one, e.g. the First Come First Served discipline.

For the token ring local area network mentioned above the time for the token to be passed from station to station is in general not negligible. Correspondingly, in the polling model the time the server needs to switch from station to station has to be taken into account. This fact considerably complicates the analysis: the principle of work conservation is no longer valid, since now the server may be idle (switching), although there is at least one customer in the system. However, under certain conditions there exists a natural modification of the principle of work conservation for polling systems with switch-over times, based on a *decomposition* of the amount of work in the system. This result, which has been proved at CWI, states that - under certain conditions - the amount of work in the polling system, $\mathbf{V}_{with}$, is in distribution equal to the sum of the amount of work in the simpler 'corresponding' system *without* switch-over times, $\mathbf{V}_{without}$, plus the amount of work, $\mathbf{Y}$, at an arbitrary moment during a period in which the server is switching from one queue to another:

$$\mathbf{V}_{with} = \mathbf{V}_{without} + \mathbf{Y} \tag{2}$$

The work decomposition gives rise to similar expressions for a weighted sum of the mean waiting times as Formula (1). Taking means in (2) leads to (cf. (1))

$$\sum_{i=1}^{N} \rho_i E\mathbf{W}_i = \rho \frac{\sum_{i=1}^{n} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + E\mathbf{Y}. \tag{3}$$

Denote the mean total switching time in one round of the server by s, and the second moment by $s^{(2)}$. Evaluating $E\mathbf{Y}$ yields:

$$\sum_{i=1}^{N} \rho_i E\mathbf{W}_i = \rho \frac{\sum_{i=1}^{N} \lambda_i \beta_i^{(2)}}{2(1-\rho)} + \rho \frac{s^{(2)}}{2s} + \frac{s}{2(1-\rho)}[\rho^2 - \sum_{i=1}^{N} \rho_i^2] + \sum_{i=1}^{N} E\mathbf{M}_i, \tag{4}$$

where $E\mathbf{M}_i$ denotes the mean amount of work in Q_i left behind by the server upon its departure from that queue (when $s \rightarrow 0$, the fraction of visits to Q_i in which the server finds Q_i empty tends to one, and the right-hand side of (4) reduces to the right-hand side of (1)). Formula (4) has been coined a *pseudo-conservation law.* The main difference with Kleinrock's conservation law is that now the left-hand side of the formula does depend on the service discipline at each queue, through $\Sigma E\mathbf{M}_i$. For many service disciplines, including exhaustive, gated and 1-limited service, we are able to determine the right-hand side of (4) explicitly. Often, such pseudo-conservation laws contain the only information available in polling systems with nonzero switching times. Hence, they are of considerable practical importance. One of the main features of the pseudo-conservation laws is that they are very useful for testing *and* developing approximations for the individual mean waiting times, which can seldom be determined explicitly.

Last year's results

Boxma and Groenendijk, in close cooperation with H. Levy (Tel-Aviv University) and J.A. Weststrate (Tilburg University), have extended the results described here in several directions.

- An accurate and generally applicable method for approximating mean waiting times has been devised, which may yield insight into the behaviour of a large class of polling systems.
- Formula (4) has been generalized to allow group arrivals. In combination with the developed mean waiting time approximation procedure, this has been used to analyze the performance of Transaction Driven Computer Systems.
- Pseudo-conservation laws have been derived for several non-cyclic server routing mechanisms. This was motivated by the consideration that cyclic routing is more and more becoming a naive strategy, dating from the days of insufficient computing power to implement anything more sophisticated.
- The ultimate goal of performance modeling and analysis is performance improvement and system optimization. Pseudo- conservation laws appear to be a good starting-point for optimization in polling systems. In the case of non-cyclic service, they have already been used to minimize mean system workload by judiciously choosing the server route. This research direction will be pursued in a forthcoming Ph.D. project.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Combinatorial optimization

Combinatorial optimization and algorithmics is the mathematical investigation of problems and algorithms involving the arrangement, grouping, ordering or selection of discrete objects. The subjects are:

- Design and analysis of algorithms (1973);
- Polyhedral methods (1983);
- Multicommodity flows and VLSI-layout (1989);
- Computational geometry (1989);
- Parallel computations (1982);
- Multi-criteria machine scheduling problems (1985);
- Interactive planning methods (1983);
- Model and algorithm representation and manipulation (1988).

A. Schrijver, A.M.H. Gerards, J.A. Hoogeveen, B.J.B.M. Lageweg, J.K. Lenstra, S.L. van de Velde, H. Oosterhout, B. Veltman.

Univ. Rotterdam, Univ. Tilburg, Bellcore, Eötvös Lorand Univ. Budapest, MIT Cambridge, Univ. Augsburg, Univ. California Berkeley, Univ. Pennsylvania, Bell Labs Murray Hill, Morristown NJ, Univ. Grenoble, Univ. Pittsburgh, Univ. Waterloo, Univ. Eindhoven, Ithaca NY, Univ. Montréal, Univ. Amsterdam, Univ. Texas.

Analysis and control of information flows in networks

The project concerns the mathematical modeling, analysis and control of information flows in computer systems and telecommunication networks. The subjects are:

- Analysis of mathematical queueing models (1981);
- Performance analysis of communication systems (1983);
- Performance analysis of computer systems (1985);
- Reliability and availability of networks (1987).

O.J. Boxma, J. van den Berg, J.L. van den Berg, J.W. Cohen, F.A. van der Duyn Schouten, W.P. Groenendijk, P. Wartenhorst.

Univ. Rotterdam, Free Univ. Amsterdam, Univ. Tilburg, Univ. Utrecht, AT&T Bell Laboratories, IBM Forschungslaboratorium Zürich, INRIA, Univ. Maryland, Univ. Eindhoven, Univ. Tel-Aviv.

System and control theory

System and control theory aims at formulating and analyzing dynamical systems as models for dynamic phenomena, and solving control and prediction problems. The subjects are:

- Deterministic system theory (1984);
- Stochastic system theory (1978);
- Systems with a generalized state space (1988);
- Control of discrete event systems (1988);
- Inverse scattering and image processing of seismic data (1988);
- Overload control for communication systems (1985).

J.H. van Schuppen, M. Hazewinkel (CWI, AM), M. Kuijper, J.M. Schumacher, P.R. de Waal (STW), J.W. van der Woude (Shell fellow).

Univ. Twente, Univ. Groningen, Philips Telecomm. Hilversum, Univ. California Berkeley, Univ. Gent, Univ. Padua.

Stochastic processes

Fundamental research on stochastic processes with special emphasis on processes in space *and* time, and research on the statistical analysis of particular stochastic processes.

- Stationary processes and their applications in physics (1981).

H.C.P. Berbee, J. van den Berg.

Univ. Delft, Univ. Bristol, Univ. Madison.

Semiparametric statistics

The construction of statistical procedures and the derivation of their properties for semiparametric models, i.e. models which are partly parametric and partly nonparametric in character. More generally, the study of statistical techniques of mixed parametric/ nonparametric nature, e.g. bootstrapping. Applications are also considered, especially in consultation and cooperative projects. The subjects are:

- Semiparametric estimation theory (1983);
- Bootstrap methods (1984);
- Statistical inference using extreme order statistics (1986);
- Statistical analysis of debugging and error-counting models in software reliability (1989);
- Nonparametric estimation of the survival function in the presence of bivariate censoring (1988);
- Statistical consultation and cooperation (1986).

K.O. Dzhaparidze, *R. Helmers*, R.D. Gill, L.F.M. de Haan, A.L.M. Dekkers, S.A. van de Geer, A.M. Shiryaev, A.W. van der Vaart, D.M. Bakker, M.C.J. van Pul, R. van der Horst (CWI-STO), B. Lisser (CWI-STO).

Univ. Baltimore, Limburg University Centre (Belgium), Univ. Tbilisi, AKZO, NLR, Philips Nat. Lab., ITI-TNO, RIVM, Gammasoft.

Image analysis

This project is concerned with mathematical and statistical aspects of the analysis of digital images and related spatial data. The aims of the project are to apply probabilistic models and statistical techniques to obtain new algorithms and assess the performance of existing ones.

- Statistical aspects of image analysis (1988).

A.J. Baddeley, R.D. Gill, R.A. Moyeed, M.N.M. van Lishout, R. van der Horst, B. Lisser.

Free Univ. Amsterdam, Univ. Amsterdam, CSIRO Australia, ITI-TNO, Univ. Aarhus.

P.J. van der Houwen (head of department)
E.D. de Goede
P.W. Hemker
W.H. Hundsdorfer
J. Kok
B. Koren
H.T.M. van der Maarel
J. Molenaar
J. Mooiman
P.A. van Mourik
R.R.P. van Nooyen
H.J.J. te Riele
J. Schlichting
R.A. Trompert
J.G. Verwer
H.A. van der Vorst (advisor)
P. Wesseling (advisor)
P.A. Zegeling

programmers:
J.G. Blom
W.M. Lioen
M. Louter-Nool
B.P. Sommeijer
D.T. Winter
P.M. de Zeeuw

trainee:
W. Couzy

STANDARDIZATION OF ELEMENTARY FUNCTIONS IN ADA

Introduction
Anyone confronted with the problem of non-fitting mechanical or electrical spares will readily admit that standardization is an important issue in society today. As programming languages emerged, so here too standards became a matter of paramount importance. In some cases standardization has been quite successful, e.g., in graphics (where CWI made important contributions to the emergence of the GKS standard) and in communication protocols.

In the field of mathematical functions, elementary functions like *sqrt*, *log* and *sin* have been available in programming languages from the beginning. However, there is a wide spectrum for which functions are available, how they can be called, and how they perform with respect to accuracy of the result and domain of the argument(s).

In the programming language Ada elementary functions are absent, for reasons mentioned below. This offered the opportunity to the people most interested in it, to define a package of elementary functions in Ada, and to give requirements for implementations of this package. The definition should satisfy Ada programming style rules, and its implementation on Ada systems should meet performance requirements coming from the numerical computing community.

In 1986 international co-operation was started to provide for such a definition. This resulted in a proposal for international standardization of a set of Ada declarations of elementary functions, together with detailed mathematical specifications and requirements. These mathematical specifications are also exemplary for other programming languages. The proposal now follows the ISO channels to achieve the addition of the proposed specifications as a standard to the Ada language.

Aspects of the proposed standard
Programming languages used for scientific computing usually define a set of standard elementary functions. For example, ALGOL60 recommended provision of *sqrt*, *sin*, *cos*, *arctan*, *ln* and *exp*. Their meaning was assumed to be obvious. Ada, however, was designed without such predefined mathematical functions. Instead, it was planned that additional packages would be developed with the best possible use of the offered language features. In terms of general characteristics, an Ada package of elementary functions, dictated by the Ada design goals, should be: readable, reliable, fault-tolerant, efficient, portable and reusable. From the mathematician's point of view such a package should provide user-friendly traditional elementary functions, with a high range of applicability, and with accuracy matching the used degree of precision.

It was soon clear that, given the language's potential, several - far from obvious - decisions could be taken. Below we summarize a few of them, together with the answers provided by the Working Groups assigned to these problems.

- Which mathematical functions should be provided, and how?
  Besides the SQRT function, a set of the most common transcendental functions has been chosen, including one operator (** for exponentiation).
- How can trigonometric functions be provided when the units of the arguments are degrees, where the period of *sin* is 360, or bams (binary angular measure), where the period of *sin* is 1, for example, instead of radians?

  The *overload* possibility in Ada allowed the solution to supplement the traditional one-argument functions like SIN(X) for the elementary function *sin* (x) (period is 2π) with two-parameter versions SIN(X, CYCLE) where the second parameter indicates the alternative period.
- Ada offers a feature to define floating-point types of a specified precision (and range). Should (a) the elementary functions be made exploitable for every user-defined floating-point type, and (b) a user with a low-precision floating-point type obtain elementary functions with low-precision and higher execution speed?
  The answer on (a) was: yes, and on (b): not always.
- What accuracy can be required for each elementary function, depending on the available floating-point precision and (usually) the value of the argument in a function call?
  The elementary functions are in all cases approximated by (piece-wise) *rational* functions, which means that the computation of a function value involves a finite number of floating-point operations $(+,-,*,/)$ with floating-point numbers. On a given interval we can always derive rational functions which stay as close as desired to the approximated function (see Figure 1). But with a chosen approximation function, one must also account for a certain round-off error made by the floating-point operations needed for computing its values. This made it possible to define for every function realistic requirements for its relative error for a wide range of its argument. In all cases the error should not be greater than a few multiples of the relative floating-point precision.
- Which mathematical inequalities can be satisfied for each function?
  It can be imagined, that for the approximation SIN of the elementary function *sin*, the property SIN(X)$\leqslant 1$ for X near $\pi/2$ is not satisfied because the approximating rational function has a different maximum (see Fig-

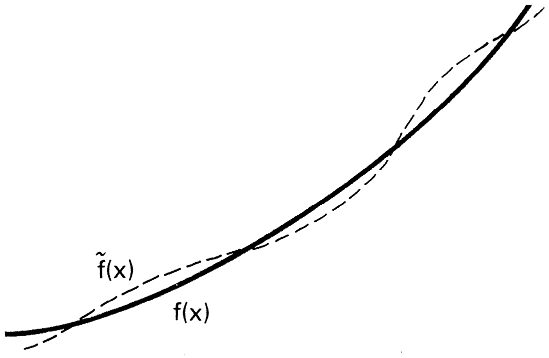


Figure 1. A function can be approximated on a given interval by (piece-wise) rational functions as close as desired.

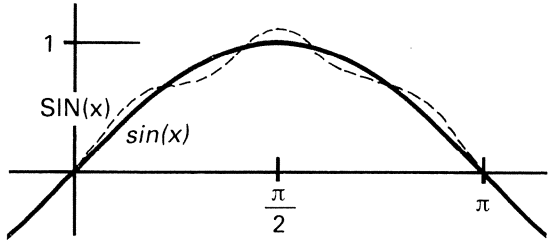


Figure 2. Approximations of a function do not always satisfy given inequalities. Here the approximation of the sin *function is larger than one near $\pi/2$. Ada working groups decided that such approximations should be excluded.*

Ada Augusta, Lady King, later Countess of Lovelace, c. 1835. (Crown copyright. The National Physical Laboratory.)
As the only legitimate daughter of the notorious romantic poet, Lord Byron, Ada Lovelace was already a well-known figure in 19th century high society at an early age. It was in this milieu that she met the computer pioneer Charles Babbage in 1833. From the start she took a keen interest in his work on calculating engines. In 1843 she translated an article by the Italian mathematician (later general and prime minister) L.F. Menabrea; its subject was the Analytical Engine, which had been occupying Babbage since 1834. In a lengthy supplement to the article Ada Lovelace explained several applications for the engine. For this she is considered the world's first programmer. Her contribution is honoured in the choice name for the programming language Ada. Alas, soon after the article was published her fortunes declined and she died poor in 1852.

The programming language Ada

In the mid-seventies the US Department of Defense (DoD) decided to do something about the babel of tongues prevailing in their software systems. By that time these systems, mostly embedded in aircraft, launching devices, etc., utilized several hundred programming languages and dialects. This led to the proposal to develop a single higher order language.

At the same time the European software industry expressed the need for a language in which to write standardized software. A study, called the European Systems Language (ESL), initiated by the Commission of the European Communities, led to the design of such a language.

In 1979 the American DoD selected one of the four competing language designs, and called it Ada, in honour of Augusta Ada, daughter of Lord Byron, who may be considered as one of the world's first programmers. Although starting from different premises, ESL's implications for the language were rather similar to those of DoD, two important desired features being portability and quality. As a consequence, the European efforts also shifted to support for Ada, and thus Ada became a truly international major development in software engineering.

Currently Ada is widely adopted as a programming language, both in an embedded environment and in scientific computing. Ada became an ANSI standard in 1983 and an ISO standard in 1987. Some major properties of the language are:

- checking of program correctness before execution;
- explicit and separate specification of interface properties of program parts;
- integration in an automated environment;
- applications with importance to IT-industry.

On the other hand, Ada still has some serious drawbacks, such as the absence of certain programming features which are much appreciated by scientific software engineers. Hopefully, such features will be added in the current Ada revision process.

ure 2). However, this would have the embarrassing effect of allowing ARCSIN(SIN(X)) to raise an exception (the Ada answer for an illegal argument of ARCSIN). Therefore, it was decided that the functions must satisfy requirements on the result like |SIN(X)|⩽1, SQRT(X)⩾0, and COSH(X)⩾1. One of these requirements, EXP(X)⩾0, could come as a surprise, since the real mathematical function *exp* (x) vanishes nowhere. But for large negative values of X the result becomes much smaller than the smallest positive floating-point number, which makes 0.0 the best approximation. The alternative was to limit the possible values of the argument X.

- Which mathematical identities must be satisfied?

  In general only the most obvious ones will be required, like SQRT(0.0) = 0.0, COS(0.0) = 1.0, and LOG(1.0) = 0.0 . Usually, the accuracy requirements for general arguments are sufficiently severe to make it likely that many other special cases will in practice also be satisfied.
- Which domains are defined for the functions?

  In general the traditional mathematical definition is followed closely, which resulted in requirements like: X⩾0 for SQRT, X>0 for LOG, and X=0 for COT. One choice that provoked much discussion was to exclude the possibility 0.0**0.0 for X**Y.

  The importance of the proposed standard here is that it does not allow implementations to limit the domain of a function more than reasonably necessary. Where the proposal says that the domain is 'mathematically unbounded', the actual region for the argument of a function is often limited, because the function value becomes too large to be representable as a floating-point number. In those cases narrower requirements have been given, requesting that implementations must deliver accurate results within the specified bounds.

Development of the standardization proposal

In the European context, a start came with the EC-funded project (1982-1984), which resulted in the *Guidelines for the design of* *large modular scientific libraries in Ada* - a joint effort by CWI and the National Physical Laboratory (Teddington). The guidelines produced contained a recommendation concerning elementary functions in Ada. This was followed closely by the present proposal. A portable implementation and testing methods were subsequently produced in a subsequent EC-funded project: *Pilot implementations of basic modules for large portable numerical libraries in Ada* (1985-1987).

Meanwhile, in 1986, two Ada working groups: the Ada-Europe Numerics Working Group and the ACM-SIGAda Numerics Working Group, had started working together to achieve an agreed proposal for the standardization of an Ada definition, and of the performance requirements for elementary function implementations. In 1988, the standardization committee ISO-IEC/JTC1/SC22/WG9 (*Ada*) appointed a Numerics Rapporteur Group (drawn from the two working groups and WG9) with the task of proposing the addition of certain mathematical facilities to the Ada language. In 1989 the Group submitted its package of elementary functions to WG9. The proposal was accepted and advanced into the ISO hierarchy.

For the numerical computing community, the development of performance standards for mathematical software, starting with the elementary functions, is a logical second step following on from successful international cooperation to obtain a standard for the performance of binary floating-point arithmetic provided by computers.

CWI actively participated in all projects and working groups mentioned above, and is represented by two staff members in the WG9's Ada Rapporteur Group. The productive and cordial cooperation is ongoing with work on proposals for other mathematical facilities like complex arithmetic, Basic Linear Algebra Subprograms (BLAS), and primitives for the manipulation of floating-point numbers (for the accurate and efficient use in elementary function implementations).
Finally, the cooperation includes work on proposals for the revision of the Ada language, and the development of testing and validation strategies concerning the proposed standard mathematical packages.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Discretization of evolution problems
Analysis, development and documentation of algorithms for the numerical solution of evolution problems for differential equations and their application to industrial problems. The subjects are:

- Stability and convergence (1978);
- Adaptive grid methods (STW) (1987);
- Static regridding methods (1989);
- 3D Shallow water equations (RWS) (1988);
- Boussinesq model (STW) (1988);
- Parallel ODE solvers (1988).

P.J. van der Houwen, J.G. Verwer, W.H. Hundsdorfer, J. Mooiman, B.P. Sommeijer, J.G. Blom, E.D. de Goede, R.A. Trompert, P.A. Zegeling.

Shell Research Amsterdam, Univ. Valladolid, Univ. Halle, Univ. Delft, Univ. Trieste, Univ. Liverpool, Hydrodynamic, Univ. Dundee, Univ. Leiden, Delft Hydraulics, Min. Public Works, RIVM.

Steady boundary value problems
Development and analysis of modern tech-

niques for the efficient numerical solution of boundary value problems. In particular the study of multigrid and related methods and their application to industrial problems. The subjects are:

- Defect correction and theoretical background (1978);
- Singularly perturbed boundary value problems (1978);
- Adaptive methods (1987);
- Applications in fluid dynamics (1983);
- Efficient techniques for the solution of the Euler and incompressible Navier-Stokes equations (1983);
- Reliable and efficient methods for semiconductor device simulation equations (1987).

P.W. Hemker, B. Koren, H.T.M. van der Maarel, J. Molenaar, R.R.P. van Nooyen (IOP), P. Wesseling (advisor), P.M. de Zeeuw (STW).

NLR, Fokker, Univ. Delft, Philips CAD-Centre Eindhoven, Univ. Michigan, GMD St. Augustin, INRIA Sophia-Antipolis, Techn. Univ. Hamburg-Harburg, NAG Downers Grove (USA), Tech. Univ. Vienna, Univ. Bari, Free Univ. Brussels, Von Karman Inst. Rhode St. Genese (Belgium).

Numerical software

1. *Development of numerical software in the programming language Ada.* (1981)
2. *Vector and parallel algorithms.*
   Study of existing and development of new numerical algorithms in order to exploit the special features of vector and parallel computers. Development and production of numerical software for vector and parallel computers (in particular NEC SX-2, CRAY X/MP, Alliant FX/4, transputer systems). (1984)
3. *Computational number theory.*
   Study of how vector and parallel processors can be used in an optimal way for the solution of those number-theoretical problems where modern computers and numerical techniques can play a vital role. (1976)

J. Kok (1), *H.J.J. te Riele* (2,3), H.A. van der Vorst, E.D. de Goede, B.P. Sommeijer, W.M. Lioen, M. Louter-Nool, D.T. Winter.

1. Argonne NatLab, NAG Oxford, NPL Teddington, Univ. Amsterdam, Ada-Europe Numerics WG.
2. Univ. Trieste, Univ. Amsterdam, Argonne NatLab, Shell Research Amsterdam, CERFACS Toulouse.
3. Gesamthochschule Wuppertal, City Univ. New York, Fargo ND, Univ. Belgrado, Univ. Toronto, Australian Nat. Univ. Canberra, Bell Labs, Kent State Univ.

Department of Software Technology

J.W. de Bakker (head of department)
K.R. Apt
J.C.M. Baeten
J.A. Bergstra (advisor)
M. Bezem
F.S. de Boer
R.N. Bol
D.J.N. van Eijck
A. Eliëns
P. van Emde Boas
L.C. van der Gaag
R.J. van Glabbeek
J.F. Groote
J. Heering
P.R.H. Hendriks
E. Horita
J.M. Jacquet
P. Klint
J.W. Klop
J.N. Kok
J.W.C. Koorn
L. Kossen
K. Kunen
P.J.F. Lucas
E.A. van der Meulen
A. Middeldorp
E.R. Olderog
A. Ponse
J. Rekers
J.J.M.M. Rutten
J.W. Spee
F.W. Vaandrager
F.-J. de Vries
H. Walinska
W.P. Weijland

trainees:
D. Turi
J.H.A. Warmerdam

PROCESS ALGEBRA

Introduction

Process algebra is the study of concurrent communicating processes in an algebraic framework. The initiator of this field is considered to be R. Milner, with his *Calculus of Communicating Systems* [5], which formed the basis for most of the axiom systems in the theory ACP (Algebra of Communicating Processes) of Bergstra and Klop [2, 3]. The endeavour of process algebra is to treat concurrency theory (the theory of concurrent communicating processes) in contrast to other, less formal approaches, in an *axiomatic* way. This approach is also used , for instance, in the study of mathematical objects as groups or fields by starting with an axiomatization of the intended objects. The axiomatic method which concerns us, is algebraic in the sense that we consider structures which are models of some set of (mostly) equational axioms; these structures are equipped with several operators. Thus, we use the term *algebra* in the sense of model theory.

What is a process?

There is ample motivation for such an axiomatic-algebraic approach to concurrency theory. The main reason is that there is no single definite notion of *process.* There is a staggering amount of properties which one may or may not attribute to processes; there are dozens of views (*semantics*) which one may have on (a particular kind of) processes; and there are infinitely many models of processes. Thus an attempt to organize this field of process theories leads very naturally and almost unavoidably to an axiomatic methodology; and a curious consequence is that one has to answer the question 'What is a process?' with the seemingly circular answer 'A process is something that obeys a certain set of axioms ... for processes'. The axiomatic method has proven effective in mathematics and mathematical logic - and in our opinion it has its merits in computer science as well, if only for its organizing and unifying power.

Calculus

Alongside the organizing role of this set-up with axiom systems, their models and the study of their relations, we have the obvious *computational* aspect. Even more than in mathematics and mathematical logic, in computer science it is *algebra* that counts. For instance, in a system verification the use of transition diagrams (as for example Petri nets) may be very illuminating; but, particularly when it comes to larger systems, it is desirable to have a formalized mathematical language at our disposal in which specifications, computations and proofs can be given in what is in principle a linear notation. Only then can we hope to succeed in attempts to mechanize our dealings with the objects of interest. In our case the mathematical language is algebraic - with basic constants, operators to construct larger processes, and equations defining the nature of the processes under consideration. (However, the format of pure equations is not always enough. On occasion, conditional equations and some infinitary proof rules are used.) To be specific: we will always insist on the use of congruences, rather than mere equivalences in the construction of process algebras; this in order to preserve the purely algebraic format.

A further advantage of the use of the axiomatic-algebraic method is that the entire apparatus of mathematical logic and the theory of abstract data types is at our disposal. One can study extensions of axiom systems, homomorphisms of the corresponding process algebras, and one can formulate exact statements as to the relative expressibility of some process operators (non-definability results).

Extensions

Of course, the present axiomatizations for concurrency theory do not cover the entire spectrum of interest. Several aspects of processes are as yet not well treated in the algebraic framework. The most notable examples concern the real-time behaviour of processes, and what is called *true concurrency* (non-interleaving semantics). Algebraic theories for these aspects are under development at the moment.

Linear vs. branching time

In our view, process algebra can be seen as a worthy descendant of 'classical' automata theory as it originated three or four decades ago. The crucial difference is that nowadays one is interested not merely in the set of execution traces (or language) of one automaton, but in the behaviour of systems of *communicating* automata. Thus, as both Milner and Hoare [4] discovered, for several purposes it is no longer sufficient to abstract the behaviour of a process to a language of execution traces. Instead, one has to work with a more discriminating process semantics, in which the *timing of choices* of a system component is also taken into account. Mathematically, this difference is very sharply expressed in the equation $x \cdot (y+z) = x \cdot y + x \cdot z$, where $+$ denotes choice and $\cdot$ is sequential composition; x,y,z are processes. If one is interested in languages of execution traces (trace semantics), this equation holds; but in process algebra it will in general not hold. Nevertheless, process algebra retains the option of adding the equation and studying its effect. In fact, one goal of process algebra is to form a uniform framework in which several different process semantics can be compared and related. One can call this *comparative concurrency semantics*.

We bring structure in our theory of process algebra by *modularization*, i.e. we start from a minimal theory (containing only the operators $+$ and $\cdot$), and then we add new features one at a time. This allows us to study features in isolation, and to combine the modules of the theory in different ways.

$x + y = y + x$		A1
$(x + y) + z = x + (y + z)$		A2
$x + x = x$		A3
$(x + y)z = xz + yz$		A4
$(xy)z = x(yz)$		A5
$x + \delta = x$		A6
$\delta x = \delta$		A7
$a \mid b = \gamma(a,b)$	if γ defined	CF1
$a \mid b = \delta$	otherwise	CF2
$x \parallel y = x \mathbin{\|\!\!\!\_} y + y \mathbin{\|\!\!\!\_} x + x \mid y$		CM1
$a \mathbin{\|\!\!\!\_} x = ax$		CM2
$ax \mathbin{\|\!\!\!\_} y = a(x \parallel y)$		CM3
$(x + y) \mathbin{\|\!\!\!\_} z = x \mathbin{\|\!\!\!\_} z + y \mathbin{\|\!\!\!\_} z$		CM4
$ax \mid b = (a \mid b) \cdot x$		CM5
$a \mid bx = (a \mid b) \cdot x$		CM6
$ax \mid by = (a \mid b) \cdot (x \parallel y)$		CM7
$(x + y) \mid z = x \mid z + y \mid z$		CM8
$x \mid (y + z) = x \mid y + x \mid z$		CM9
$\partial_H(a) = a$	if $a \notin H$	D1
$\partial_H(a) = \delta$	if $a \in H$	D2
$\partial_H(x + y) = \partial_H(x) + \partial_H(y)$		D3
$\partial_H(xy) = \partial_H(x) \cdot \partial_H(y)$		D4

Algebra of Communicating Processes (ACP).

$$
\begin{array}{ll}
a \xrightarrow{a} \surd & \\
x \xrightarrow{a} x' & \Rightarrow x+y \xrightarrow{a} x' \text{ and } y+x \xrightarrow{a} x' \\
x \xrightarrow{a} \surd & \Rightarrow x+y \xrightarrow{a} \surd \text{ and } y+x \xrightarrow{a} \surd \\
x \xrightarrow{a} x' & \Rightarrow xy \xrightarrow{a} x'y \\
x \xrightarrow{a} \surd & \Rightarrow xy \xrightarrow{a} y \\
x \xrightarrow{a} x' & \Rightarrow x \| y \xrightarrow{a} x' \| y, \\
 & \quad y \| x \xrightarrow{a} y \| x' \text{ and } x \mathbin{\lfloor\!\lfloor} y \xrightarrow{a} x' \| y \\
x \xrightarrow{a} \surd & \Rightarrow x \| y \xrightarrow{a} y, \\
 & \quad y \| x \xrightarrow{a} y \text{ and } x \mathbin{\lfloor\!\lfloor} y \xrightarrow{a} y \\
x \xrightarrow{a} x',\ y \xrightarrow{b} y' \text{ and } \gamma(a,b) = c & \Rightarrow x \| y \xrightarrow{c} x' \| y' \text{ and } x | y \xrightarrow{c} x' \| y' \\
x \xrightarrow{a} x',\ y \xrightarrow{b} \surd \text{ and } \gamma(a,b) = c & \Rightarrow x \| y \xrightarrow{c} x',\ x | y \xrightarrow{c} x', \\
 & \quad y \| x \xrightarrow{c} x' \text{ and } y | x \xrightarrow{c} x' \\
x \xrightarrow{a} \surd,\ y \xrightarrow{b} \surd \text{ and } \gamma(a,b) = c & \Rightarrow x \| y \xrightarrow{c} \surd \text{ and } x | y \xrightarrow{c} \surd
\end{array}
$$

Operational Semantics of ACP.

Projects

Process algebra is studied at CWI in cooperation with the University of Amsterdam (Programming Research Group), and the University of Utrecht (Applied Logic Group). To a large degree, the theory was developed in the framework of the ESPRIT I project METEOR, (ended 1 October 1989). Currently, the theory is used in three European projects:

- The theory will be further developed in the ESPRIT Basic Research Action CONCUR, and close cooperation will be started with the originators of other algebraic concurrency theories, viz. CCS, CSP and MEIJE.
- In the RACE project SPECS, process algebra forms the basis for the specification language CRL (Common Representation Language), into which specifications in 'tower languages' such as LOTOS, SDL and CHILL are translatable, and which in turn is the starting point for derived representations and tools.
- In the ESPRIT II Technology Integration Project ATMOSPHERE, process algebra will be used to verify specifications written in SDL.

References

1. J.C.M. Baeten (ed.) (1990). *Applications of Process Algebra,* Cambridge University Press.
2. J.A. Bergstra, J.W. Klop (1984). Process algebra for synchronous communication. *Inf. & Control 60* (1/3), 109-137.
3. J.A. Bergstra, J.W. Klop (1985). Algebra of communicating processes with abstraction. *Theor. Comp. Sci. 37* (1), 77-121.
4. C.A.R. Hoare (1985). *Communicating Sequential Processes,* Prentice- Hall.
5. R. Milner (1980). *A Calculus of Communicating Systems,* Springer LNCS 92.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Concurrency
Research into the semantic aspects of parallel computation according to various programming styles (imperative, applicative, logic, object-oriented); also foundational topics related to semantic modelling. (1984)

J.W. de Bakker, E. Horita, J.M. Jacquet, J.N. Kok, J.J.M.M. Rutten, J.H.A. Warmerdam.

ESPRIT BRA-partners, REX-partners, Free Univ. Amsterdam, Univ. Utrecht, Univ. Rotterdam.

Formal specification methods
Research concerning process algebra, specification languages, executable specifications and systems development methodology. (1982)

J.C.M. Baeten, J.A. Bergstra, R.J. van Glabbeek, J.F. Groote, A. Ponse, F.W. Vaandrager, W.P. Weijland.

ESPRIT-partners, RACE-partners, Univ. Amsterdam, Univ. Utrecht, Philips Research Eindhoven, PTT Research Leidschendam.

Extensible programming environments
Algebraic specification of programming environments, incremental development of language definitions, implementation of algebraic specifications. (1982)

P. Klint, J. Heering, P.R.H. Hendriks, J.W.C. Koorn, E.A. van der Meulen, J. Rekers.

ESPRIT-partners, Univ. Amsterdam, SERC Utrecht.

Term rewriting systems
Foundational research centering around term rewriting systems, with an emphasis on algebraic and syntactic methods; foundational research in process algebra. (1989)

J.W. Klop, R.J. van Glabbeek, A. Middeldorp, F.-J. de Vries.

ESPRIT BRA-partners, Free Univ. Amsterdam, Univ. Amsterdam, Univ. Utrecht, Univ. Nijmegen, Univ. Leiden.

Expert systems
Research into the applicability of methods of knowledge representation and automated reasoning in expert systems. The development of prototype expert system tools. (1985/1986)

P.J.F. Lucas, A. Eliëns, P. van Emde Boas, L.C. van der Gaag, L. Kossen, J.W. Spee.

Philips Research Eindhoven, Univ. Twente, Univ. Amsterdam.

Logical aspects of artificial intelligence
Research concerning logic programming, deductive and knowledge-based database systems, theorem proving, non-monotonic reasoning and natural language processing.

K.R. Apt, M. Bezem, P. van Emde Boas, D.J.N. van Eijck, H. Walinska, R.N. Bol, K. Kunen.

Univ. Syracuse, Univ. Pisa, Univ. Victoria (Canada), Free Univ. Amsterdam, Univ. Amsterdam.

L.G.L.T. Meertens (head of department)
L.G. Barfield
C.A. van den Berg
H. den Boer
E.D.G. Boeve
J.N.E. Bos
P. Bosch
D. Chaum
M.J. Coster
M.M. Fokkinga
E.J.L.J. van Heyst
J.T. Jeuring
M.L. Kersten
E. Kranakis
D.D.M. Krizanc
S.J. Mullender
S. Pemberton
G. van Rossum
F.H. Schippers
I. Shizgal
A.P.J.M. Siebes
J.T. Tromp
N.Th. Verbrugge
P.M.B. Vitányi
M.H. van der Voort

programmers:
C. Algeo
C.J. van Antwerpen
H.G.P. Bosch
F. van Dijk
A.J. Jansen
T.J.G. Krijnen
S. van der Zee

trainees:
P. Heynen
H.E. Lohuis
H.J. Meinders
R. Preuter
P.G.C.M. Sliepenbeek
E. Sneekes
A. van der Vuurst
A.J. de Witte
R.M. Yntema

USER INTERFACES: THE VIEWS PROJECT

Introduction

As computers have become more powerful, and the emphasis has moved from mainframes to personal computers, so the emphasis in software has moved from batch to interactive programs. Accompanying this change of emphasis, attention has focused on the *user-interface* of programs: the manner in which one works with a program, and what it is possible to do with it.

And as the number of interactive programs has increased, the users have been faced with an ever increasing number of different user-interfaces, so that to accomplish some task (such as sending out electronic mail) you may have to change user-interface several times, leading to confusion, errors, and inefficiency.

At CWI a project group is presently working on various aspects of user-interfaces.

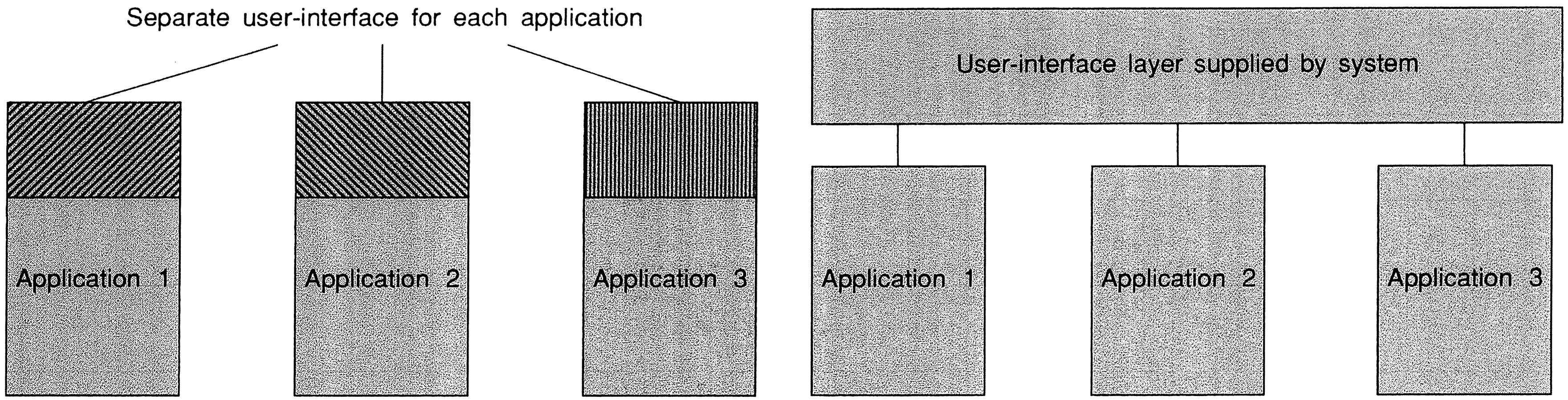
Present situation

In order to address some of these problems, at various places different approaches have been tried. The most prevalent is the 'toolkit plus guidelines' approach: the system supplies a 'toolkit' of routines that a programmer can call from a program, and a set of guidelines on how and where to use them. The toolkit may contain, for example, ways to build menus, to open windows, etc.

An exemplary instance of this is the Apple Macintosh computer, one of the better examples of a unified approach to user-interfaces. But even with this system, user-interfaces have to be written separately for each program. Two of the very first applications available - the Desktop and Macpaint - had divergent user-interfaces: although having a common starting point, the results were rather different, for example in their window management. Generally speaking, individual programs retain their own peculiarities. The underlying problem is that while toolkits bring the user-interfaces of programs closer to each other, they can't guarantee that they will be applied in the same way.

A better approach is a user-interface layer above the whole system, so that the individual applications are unaware of user-interface issues. This 'interface independence' (or 'Dialogue Independence', [ref. 4]) can be likened to the 'I/O independence' of the 1960's, where the responsibility of device-dependent input-output was taken out of the hands of individual programs, and placed in a central kernel of the system to which all programs had access.

Another good example of this idea is file-name expansion. This is where you can give to a program a template pattern of the file



In current systems, each application has to implement its user-interface. Even on systems where a toolbox of user-interface tools is supplied, user-interfaces of different applications often differ, interfering with a user's optimal use of the computer. In Views, the user-interface is a separate layer supplied by the system, guaranteeing consistency across applications.

names you want, and the system expands this pattern into the true list of file names before passing it onto the program. For example, the print command *lprint*.doc* refers to all programs ending in *.doc* (the template pattern). On systems where this expansion facility is part of the system (such as Unix), all programs have the advantage of the facility without having to know anything about it. On systems where the facility is only part of the toolkit (such as MS-DOS), the only programs that have the facility are those where the individual programmer has taken the trouble to use it.

Another problem with current systems is a *lack of integration* between applications. Again drawing on the Apple Macintosh as a well-known example, it pioneered multi-media applications, where for instance you can include text and pictures in a single document. But you have to use different programs for producing pictures and producing text, and it makes a difference whether you create a text with drawings in it, or a drawing with text in it, and, in producing the complete document, you have to change context to produce the drawing that you want to include in your text (or vice versa). Finally, once you have included the drawing in your text, it is 'frozen': there is no chance to make further changes to it.

Views

The Views project has developed from an earlier project at CWI, the ABC Programming Language [ref. 1]. The purpose of ABC is to offer a powerful programming language and environment that is extremely easy to learn and use. As part of this aim, we wanted to reduce the number of different 'faces' that the user would see from the system: traditionally when programming, you must learn the command language of the computing system you are using, the command set of an editor, the programming language itself, how to use the compiler, and so on. With ABC, we managed to reduce this set to two: the *language* and the *editor*. ABC is not only the programming

language, but also the command language, and for tasks that are not carried out with the ABC language, you use the editor: this includes actions like renaming and deleting objects, which is done by 'direct manipulation' of the objects involved [ref. 2]
It was quickly obvious that we could extend this method of working to more general tasks than programming alone, e.g., document management and process management, and thus was born the Views project.
Views, then, provides the user with a computing environment, which is characterized by

- what-you-see-is-what-you-get (WYSIWYG),
- direct manipulation,
- an open architecture,
- a consistent interface across applications.

WYSIWYG refers to systems where the system tries to keep the screen up-to-date with the true state of things. In fact Views is slightly stronger, leading us to coin a new term: TAXATA - things are exactly as they appear. While usually in WYSIWYG systems you are working with a copy of the object in some buffer (in other words, what you see is what you *will* get), in Views you are always effectively working with the object itself.

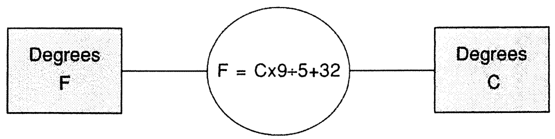
Open architecture means that it has to be easy to add new applications to the system, and that the user has much control and choice over individual aspects of the system and its interface (such as changing menus self, choosing which keys are available for 'short cuts', etc.).

User model
By the *user model* of a system we mean the model or set of rules which the user forms when working with the system. For example, if you work on a file, do you work on the file itself or on a copy? The basic user-model in Views is: *every object in the system is editable, and every action is carried out by editing.* Objects can be for example files, the clock, or menus. Traditionally, editing a file is performed on a copy, which then has to be explicitly written back to the disk. The Views model is closer to what one does in everyday life when changing a document.

As an example of the basic model, consider document management. Instead of individual commands to list the documents that you have in a directory or folder, to rename them, to delete them, copy them, and so on, you just 'visit' the folder - which is a document in itself in Views. This causes its contents to be displayed on the screen. To rename a document in the folder you just edit its name; to delete it, you just delete its entry; to copy a document, you just use the normal copy facility of the editor.
Similarly, to read electronic mail you just 'visit' your mail box. This causes its contents to be displayed as a list of message headers. Again, you can visit these individual messages (themselves documents), rename them, delete them, copy them, all in exactly the same way.

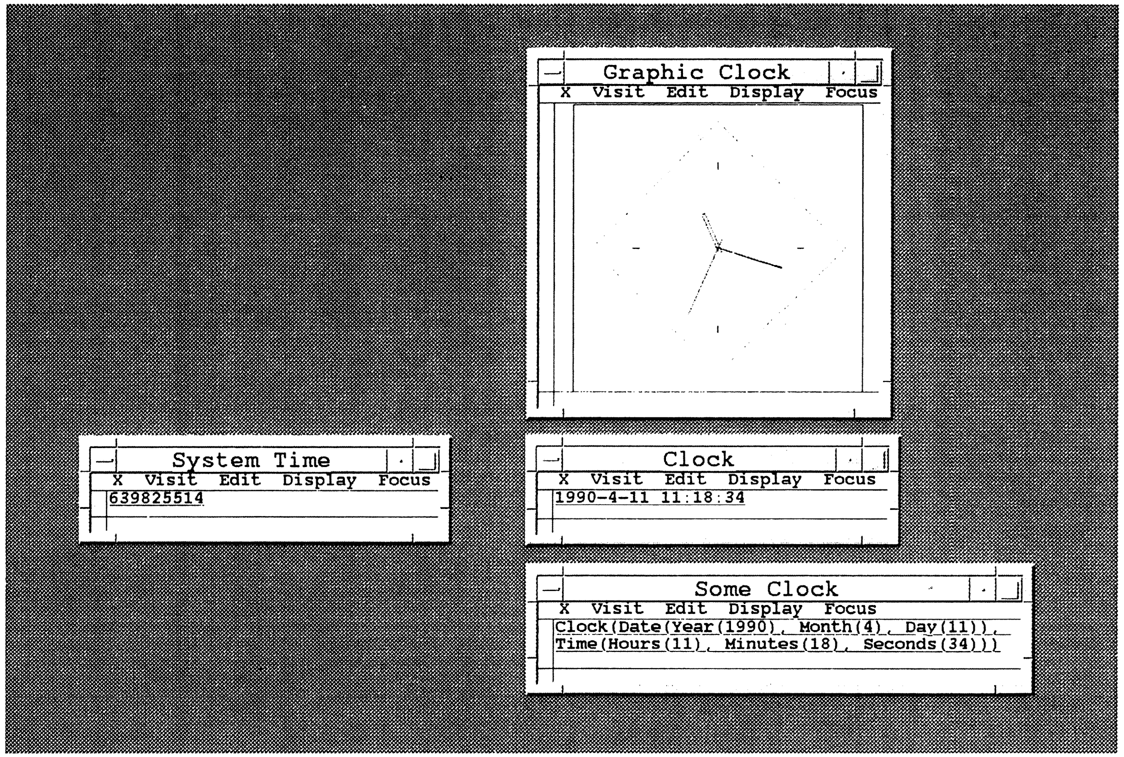
An so on for other tasks: adding or deleting things to the printer queue, listing and deleting running processes, editing textual documents, or amending a spread-sheet.

Implementation model: invariants
The main characteristic of the implementation is that in general there are invariants between objects in the system. These invariants state that the contents of an object are a function of one or more other objects. For example, the profit for this year is the difference between the income and outgoings. If an object gets changed (usually by the user editing it), the invariant goes 'out-of-date', and has to be re-instated, which is done by calling a related function.



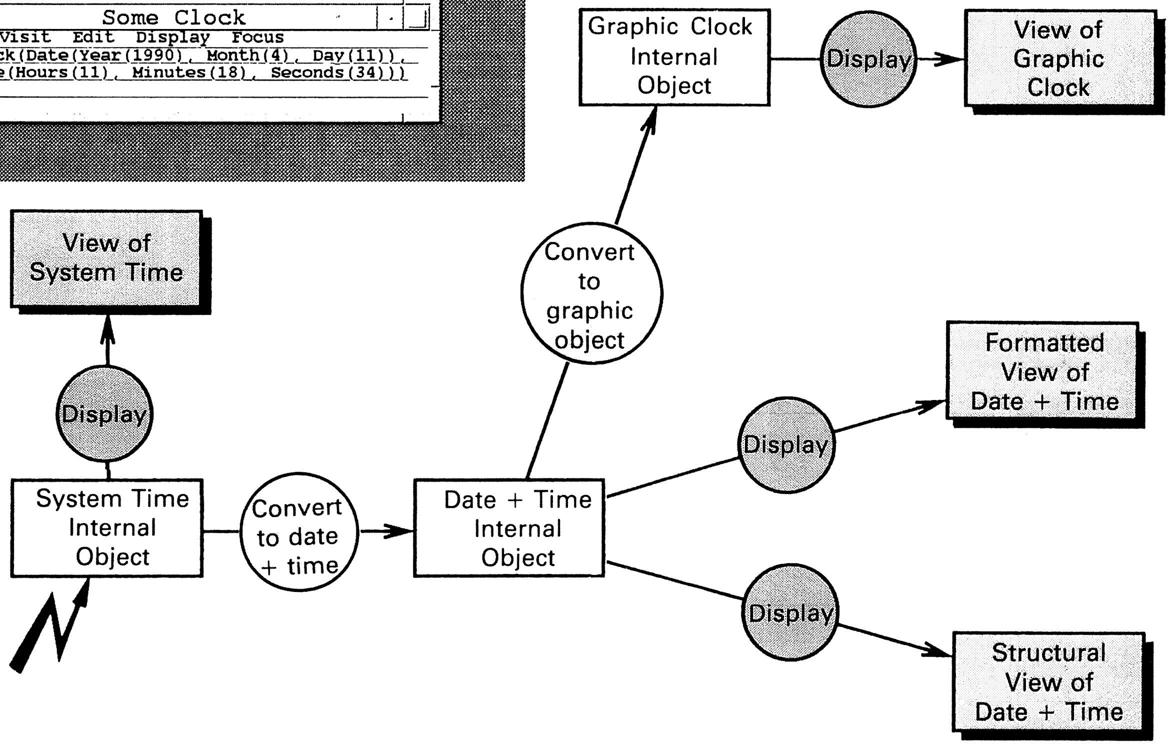
Invariants, here represented as a circle, link objects together. If an object is changed, the invariant is reinstated by changing linked objects. Here an object representing the temperature in degrees Fahrenheit is linked by an invariant to an object representing degrees Celsius. If the user edits either one, the other gets updated automatically.

In fact, this mechanism is used very generally throughout the system, so that for instance, displaying objects on the screen is done by application of the invariant 'the representation on the screen must match the object': if



△
Figure 1. Four views of the time. On the left is the time as represented internally by the computer: the number of seconds since the beginning of 1970. This value is then projected in three different ways: as an analogue-style display of the time, as a digital-style display, and lastly showing the internal structure of the digital-style display, without any added formatting. In this last display, the type of each (sub-) object is given, followed by its value in brackets. Each display also shows its name, and has a 'menu-bar' for editing the object and controlling the display.

Figure 2. A simplified view of the internal connections of the objects displayed in Figure 1. The oblongs represent objects, and the circles functions connecting them. If an object changes, any connected functions are called to update connected objects. The system-time object gets changed automatically each second by the system.
The diagram has been simplified to show the main relationships, but the display functions, for instance, are also connected to objects describing how these are to be displayed, and giving the details (size, etc.) of the windows in which they are to be displayed. ▽



the object gets edited, then the screen gets updated. This means that the rest of the system can be completely oblivious of anything to do with output to the screen, or even that it occurs at all. In fact, all that an individual application sees is that its objects somehow change, and that it has to re-instate the invariants.

Present achievements

In 1989, the basic Views kernel was constructed. The idea is to construct the system incrementally, so that at all times we have a running system, and to add applications one by one.

The window interface was constructed on the basis of an earlier product of the ABC project, STDWIN, a window management package that allows programs that use windows to be portable between different windowing systems [ref. 3]. For example, STDWIN already runs on top of X-windows on Unix, on the Apple Macintosh, and on the Atari ST.

A basic object can be represented in various ways. The basic display mechanism, which displays objects on the screen, was constructed for graphical objects as well as for textual ones, and the fundamental data-types as well as some primitive editing actions were implemented.

After these necessary preliminary steps, the invariant mechanism - probably the central part of the system - was built, and a start made on some applications, such as basic text-editing, file browsing, and message reading.

References

1. L. Geurts, L.G.L.T. Meertens, S. Pemberton (1990). *The ABC Programmer's Handbook.* Prentice-Hall, ISBN 0-13-000027-2.
2. J. van de Graaf (1984). *Towards a Specifiaction of the B Programming Environment.* CWI report CS-R8408.
3. G. van Rossum (1988). *STDWIN - A Standard Window System Interface.* CWI Report CS-R8817.
4. H. Rex Hartson, Deborah Hix (1989). Human-computer interface development: concepts and systems for its management. *ACM Computing Surveys, Vol. 21, 1,* March 1989.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Algorithms and complexity

The design and analysis of algorithms in distributed computing and VLSI. Fundamental studies and research in complexity theory. (1980)

P.M.B. Vitányi, E. Kranakis, D.D.M. Krizanc, J.T. Tromp.

Univ. Amsterdam, Univ. Utrecht, Univ. Rochester, MIT, Harvard, York Univ., North-Eastern Univ., Boston Univ., Boston College, Technion, Patras Comp. Tech. Inst., Univ. Waterloo, Bell Labs.

Distributed systems

Research on performance, transparency, fault tolerance and scalability in distributed systems. Research themes for the next five years are multimedia support (voice and video) and applications in distributed systems, in particular persistent object storage, user interfaces for distributed and parallel applications, design of a multiprocessor kernel, structuring fault tolerance in replicated services. (1984)

S.J. Mullender, C. Algeo, P. Bosch, A.J. Jansen, G. van Rossum, I. Shizgal, S. van der Zee.

Free Univ. Amsterdam, Univ. Twente, Univ. Amsterdam.

Computer systems and ergonomics
This project is concerned with the methodology of integration of functions and applications in computer systems in order to provide end-users with easily manageable tools. There are three subprojects:

- ABC. The aim of this project is the design, implementation and distribution of a simple, structured, interactive programming language, built in into an integrated environment and meeting the requirements of modern personal computing. (1975)
- Human-computer interfaces (Views). The design and pilot implementation of a direct-manipulation, open-architecture, WYSIWYG, computing environment, where consistency of user-interface is guaranteed across applications. (1988)
- Euromath. The aim is to link mathematicians in (currently) 19 European countries via a communications network, allowing them to access remote databases, edit and format mathematical articles, and communicate via conferencing systems and electronic mail; and to provide a system with a single user-interface to allow the mathematicians to access and control these functions. (1987)

S. Pemberton, L.G. Barfield, E.D.G. Boeve, F. van Dijk, T.J.G. Krijnen, H.E. Lohuis, L.G.L.T. Meertens.

Univ. Utrecht, Univ. Groningen, Euromath-partners (INRIA, FIZ Karlsruhe, Univ. Paderborn, Univ. Bath).

Distributed adaptive information systems
Research on data models for adaptive information systems, programming languages for their applications, and their associated machine architectures; in particular the object-oriented approach for database modelling, the development and the evaluation of object-oriented database language concepts, and the development of software prototypes on (virtual) machines especially designed for an object-oriented programming environment. (1985)

M.L. Kersten, C.A. van den Berg, F.H. Schippers, A.P.J.M. Siebes, N.Th. Verbrugge, M.H. van der Voort.

Univ. Amsterdam, Univ. Twente, Philips Research Eindhoven.

Constructive algorithmics
The development of concepts, notations, formalisms and methods, on a high level of abstraction, for deriving algorithms from a specification. The aims include the unification of specification formalisms and formalisms for denoting algorithms, and the development of specialized theories for certain data types or classes of problems. (1977)

L.G.L.T. Meertens, M.M. Fokkinga, J.T. Jeuring, S. Pemberton.

Univ. Groningen, Univ. Nijmegen, Univ. Utrecht, Univ. Oxford.

Cryptology
All aspects of cryptology and information security are covered. Particular emphasis is given to the construction and analysis of cryptographic algorithms and protocols, practical implementations, and related complexity-theoretic investigations.

D. Chaum (NFI), H. den Boer (NFI), J.N.E. Bos, M.J. Coster (NFI), E.J.L.J. van Heyst, A.G. Steenbeek (CWI-STO).

PTT Research Eindhoven, Philips USFA, Univ. Leiden, INRIA, Aarhus Univ., Siemens, GMD, Gretag, Univ. Leuven, Univ. Montréal, Univ. Tel-Aviv, Univ. Toronto, IBM Almeden, Technion Haifa.

Department of Interactive Systems

P.J.W. ten Hagen (head of department)

F. Arbab	D.B.M. Otten	*programmer*:
E. Blake	J.L.H. Rogier	R. van Liere
C.L. Blom	M.M. de Ruiter	M. Guravage
M. van Dijk	H.J. Schouten	
W. Eshuis	P. Spilling	*trainees*:
I. Herman	C.G. Trienekens	P. Booyen
H.E. Klarenbosch	P.J. Veerkamp	V. Sakas
A.A.M. Kuijk	J. van der Vegt	

A DATA FLOW GRAPHICS WORKSTATION

Introduction

When industry calls upon CWI expertise to assist in the development of an innovative product, time is usually at a premium, and resources are limited. Even so, prospects are good if the new ideas can be realized. Above all, a concentrated effort is required. Over the years, the department of Interactive Systems has built up considerable expertise in advanced workstation design, especially in relation to computer graphics. This led to our involvement in an intensive effort to provide a production-prototype of a graphics workstation in the mini-supercomputer class. The industrial partner was a new company, Dataflow Technology Nederland B.V. or DTN for short. Also involved as a subcontractor to CWI was Parallel Computing, a small hi-tech software house. DTN was tasked with all hardware development; CWI was responsible for the software architecture; and Parallel Computing was to provide the software tools to program the highly parallel processor system inside the workstation. In the sequel we will describe this exciting project. For CWI, involvement in such endeavours (based on a research contract) brings attractive synergetic benefits: we help create new industries, we see our own research results applied, and - just as importantly - we win further expertise.

Dataflow and Computer Graphics

Generating pictures visible on a screen from a formal description requires a vast amount of computations. The description contains a *geometry component*, for instance 'a line from $A(1,0,0)$ to $B(5.31,2,1.67)$'; a *qualitative component*, for instance 'the line is red'; and a view, for instance 'the line is seen from the point $(-10,-10,0)$, looking in the direction of the positive x-axis'. A picture may contain millions of such elements. Pictural changes may happen in real time by altering any of these descriptive components.

The combination of processing strategies used to achieve fast picture generation includes: treating many elements simultaneously; having fast special purpose hardware for typical instructions such as matrix transformations for calculating the view component effect; trying to reuse as much as possible from the previous result when there is a change, etc. The massive amounts of data for such pictures which need to flow through the calculating engine is another problem which adds to the complexity; if not treated properly it has a paralyzing effect on the performance of the total calculating power, very similar to what we know as traffic jams.

The hardware solution

The advanced hardware system designed for this purpose uses two innovative principles. One is massive parallelism based on dataflow processing. It is in fact the first hardware configuration worldwide which has managed to pass the experimental stage. Dataflow is the most flexible form of parallelism, because it can combine fine-grained parallelism (on the instruction level) with coarse-grained parallelism (concurrent execution of procedures and subprograms), and it can optimize for a balanced flow of information between all its components. The computer architects have succeeded in providing this flexibility without sacrificing the basic speed of the machine and memory cycles.

The other principle is the organization of the storage for the *current picture*, i.e. the one that is currently visible on the screen. The storage function is enhanced with highly parallel processors for changing the storage contents, shielding off parts of pictures, for instance due to their being hidden behind nearer elements. Moreover the storage function has an extremely high bandwidth parallel interface to the dataflow component.
A scheme of the hardware is given in Figure 1. Each of the components shown there can work simultaneously. In turn, many processors are concurrently active inside each component. Hence, for example, the configuration as a whole can execute about 500 M instructions per second and at the same time transport about 500 Mbyte of data per second between components. These instructions and data may be dedicated to hundreds of independent combinations of processing streams.

Figure 2 gives an idea of the principle of dataflow computing. Many computing nodes may be active in computing and passing data to logically connected nodes. However, if one looked at one particular physical node, a fraction of a second later it might be involved in computing for an entirely different process. This means that not only data, but also new instructions flow through the net to occupy a node for a while. This example is only one of many that can be given of the power and flexibility of this hardware. The question which we took as a challenge is: how can it be effectively programmed?

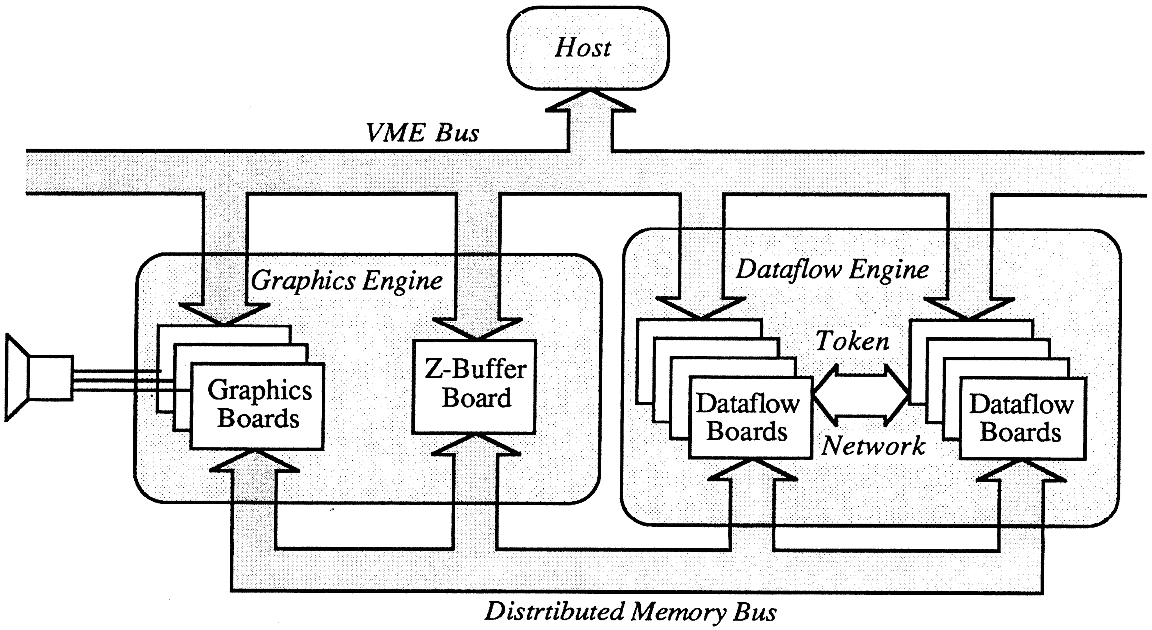


Figure 1. The Dataflow Graphics Computer. The Graphics Engine generates pixels for lines, generates filled/shaded polygons, performs image processing functions and handles a Z-buffer for automatic hidden line/hidden surface generation; the Dataflow Engine is responsible for 3D transformations. The Distributed Memory Bus provides a high-speed data communication between the Graphics and the Dataflow Engines; the Token Network is used to exchange data tokens among the Dataflow Boards themselves. The Dataflow Graphics Computer can communicate with any kind of host machine via a Standard VME bus.

The software solution

The user of the graphics software is the applications programmer who will incorporate calls to the graphics facilities in his program. Nowadays graphics functionality has been standardized. This results in several standardized calling interfaces for the applications programmer. The graphics software can choose its own methods to execute the calls using its own hardware efficiently - provided that the result is the correct function accord-

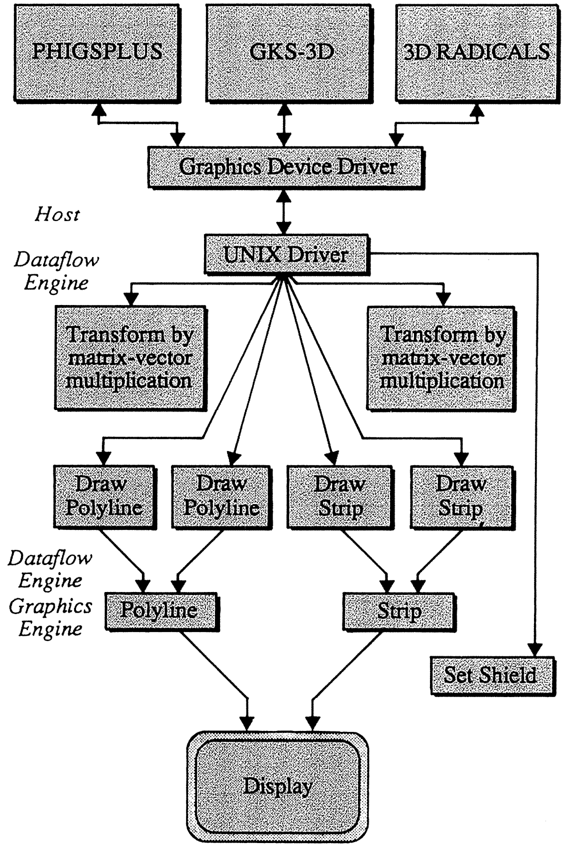


Figure 2. The dataflow software for the graphics pipeline. Two pipelines are set up to run in parallel; their output is multiplexed on the display level. The two matrix-vector multiplication subunits can run in parallel with actual drawings. Filled areas are rendered with the help of triangle strips. Shields may be set into picture memory to ensure clipping and windowing effects.

ing to the standard. For 3D graphics PHIGS, PHIGSPLUS and GKS-3D are (the) three existing standards.

In the host computer (see Figure 1) a graphics call is translated into calls to the hardware. In turn, this hardware is seen as a set of more specific graphics functions. The latter functions are passed to the hardware by the graphics device driver. At that point, control is passed to the specific hardware. In our case these are the dataflow engine and the graphics engine.

The software architecture recognizes a number of parallelisms. First of all the host and the graphics engines are simultaneously active. The host can continue to issue graphics calls without having to wait for each successive call to finish. Each call leads to a sequence of processing steps by the dataflow and graphics hardware. These steps can be pipelined. That means that all steps can execute concurrently, but for different graphics elements. Parallelism in the dataflow engine is such that several pipelines can exist next to each other, each handling a stream of graphics elements. The graphics engine accepts only one function at a time, but it accepts them at a very high rate. Hence all graphics pipelines end in a buffer containing a sequence of graphics instructions for the graphics engine.

We have thus described the coarse-grained forms of parallelism in the architecture. Typical processing steps in a pipeline are: add a graphics primitive in a list (the dataflow engine may keep many of such lists); perform a geometric transformation on a graphics primitive (e.g. matrix times vector for all points in the primitive); or, decompose a graphics primitive into instructions for the graphics engine.

The primitives in the pipeline are polylines (sequences of connected line segments) or triangular strips (sequence of triangles where successive triangles share a side). The primitives accepted by the graphics engine are individual line segments or individual triangles, together with their colour information.

The fine-grained parallelism in the dataflow engine is used to speed up each individual step. However, it is important for a balanced information flow that each step is made approximately equal in time. This balancing can also be achieved by the fine-grained parallelism.

Figure 3 illustrates a dataflow net for a matrix-vector multiplication. This block surrounded by the appropriate control can be placed one or several times in a pipeline, depending on the required throughput speed.

Dataflow programming

Dataflow programming directly on the hardware level is a difficult and tedious task. A compiler (for a subset of the C-programming language) and a debugger/simulator are available to assist the programmer. However, fine

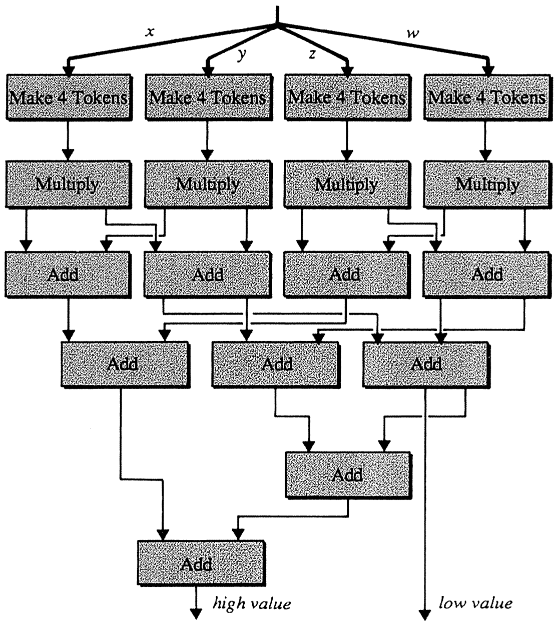


Figure 3. Matrix-vector multiplication in dataflow. The 'multiply' tokens automatically read the next element of the matrix; the four elements to be added are calculated in parallel. The high and low values of the results must be propagated separately to ensure higher accuracy. Note that while the additions are made, the multiplication nodes already perform the calculations on the next elements.

tuning for optimal performance cannot yet be done automatically; hence, optimal programming requires a specialist. However, as with many parallel machines, a reasonable compromise can be achieved using automatic tools only.

Conclusion

This joint project has produced a prototype for a competitive high performance graphics workstation. It has revealed the potentials of dataflow computing. Many ideas for further improvements have emerged both for the next dataflow architecture and for the graphics engine. Several of these are being pursued in the ongoing research.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Computer graphics

The design of functionally complete basic graphics systems, with special support for interactive use. Results to be made available, on the one hand as (contribution to) international standards, on the other hand as implementations, again with special attention to efficiency required for high quality interaction. (1980)

A.A.M. Kuijk, I. Herman, M. Bakker (CWI-STO), E. Blake (STW), P. Booyen, F.J. Burger (CWI-STO), M. van Dijk, B.P. Rouwhorst (CWI-STO), C.G. Trienekens (STW).

PTL Groningen, Philips, Univ. Twente, ISO-working group TC97/SC24, Dataflow Technology Nederland BV, Parallel Computing, GMD, INRIA Paris, Univ. Tübingen, Univ. East Anglia.

User interfaces

This project focusses on systems using graphical information. The user interface for these systems must support the construction, manipulation and interpretation of pictures. For the construction and manipulation of pictures by users a picture editor will be developed, while for the interpretation of pictures by the computer the construction process and associated semantic information will be used.

M.M. de Ruiter, *C.L. Blom*, P.J.W. ten Hagen, J. van der Vegt.

Univ. Nijmegen, Univ. Amsterdam, GMD St. Augustin.

Dialogue environments

• Dialogue programming. (1983)

The project is aimed at the development of a complete programming method for interactive dialogues. Currently a prototype system (DICE) exists which is being applied to a number of test applications. The experimental system has revealed a further set of fundamental problems which will be addressed in the next version. This will allow us to widen the scope of application to machine-machine dialogues and simplify the embedding in a variety of general purpose programming languages. Previous results concerning methodology of graphical interaction and window management will in the next few years form the basis of a new generation of international graphics standards.

P.J.W. ten Hagen, M. van Dijk, H.J. Schouten, R. van Liere, D. Soede (CWI-STO).

PTL Groningen, Univ. Twente, Philips CAD Centre, TNO, Océ, Philips, Univ. Amsterdam, Techn. Hochschule Darmstadt, Univ. Tokyo, GMD, Rutherford Labs, Univ. Southern California.

• User controlled systems. (1987)

User controlled systems are information systems for which all tasks are command driven. Meta-level commands allow the definition of new tasks. The task oriented architecture requires a high degree of integration among program libraries and databases. In addition this task oriented approach must bring about comprehensible user interfaces for very complex systems, such as CIM systems (Computer Integrated Manufacturing).

W. Eshuis, D. Soede (CWI-STO), P. Spilling.

PTL Groningen, Univ. Twente, Delft Hydraulics.

Intelligent CAD systems

The project will, through the use of AI based methods and techniques, attempt to produce CAD systems which will be more complete, integrated, and have a high quality user interface. To implement such a system a language is being developed, based on the object-oriented and logic programming paradigm. This language (IDDL, Integrated Data Description Language) has special dedicated features to encode existing and newly acquired knowledge about the design object, about the design process and about their relations. The encoding and treatment of design knowledge is studied in the context of geometric modelling, object-oriented databases, user interfaces and geometric reasoning.

F. Arbab, P.J.W. ten Hagen, J.L.H. Rogier, P.J. Veerkamp (NFI), H.E. Klarenbosch, D.B.M. Otten.

Univ. Tokyo, Computer and Automations Institute Budapest, Univ. Southern California, Univ. Delft, Univ. Amsterdam, Univ. Twente, Bilkent Univ. Ankara, Univ. Strathclyde, Univ. Edinburgh.

D.C.A. Bulterman (head of department)
P. Beertema
D.F. Karrenberg
E.S. Mullender
D.L. Draper

programmers:
R. ten Kroode
F. Kuiper
C. Orange
J. van der Steen

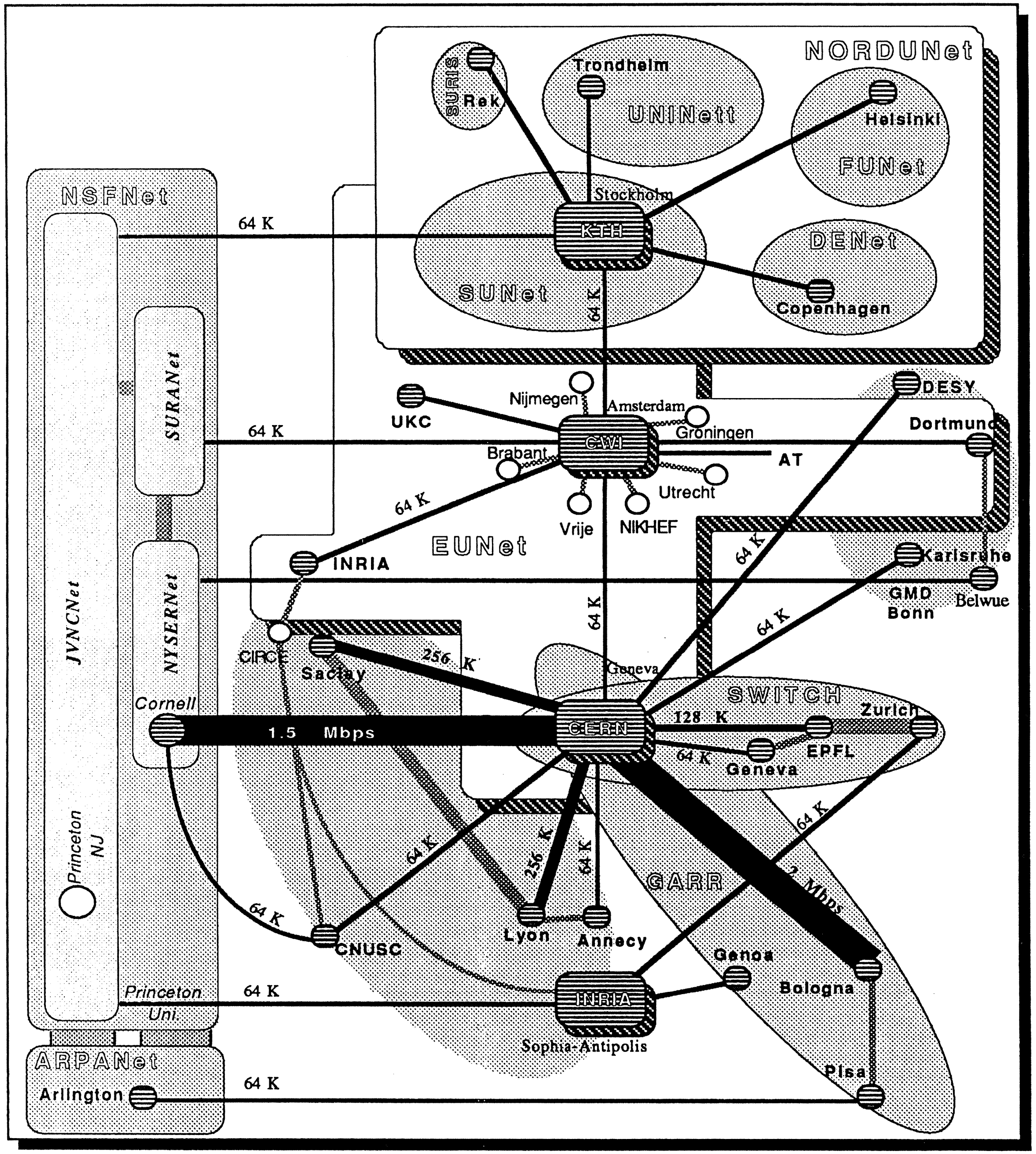
INTERNET PROTOCOL NETWORKING ACROSS EUROPE

Changes in networking

One of CWI's most visible activities during the past few years has been its involvement in international networking. As an example, consider that one of the world's more famous node addresses for users of UNIX-based communication protocols was *mcvax*, a Digital VAX computer (first a VAX-11/780, later a VAX-11/750) that was a primary link in connecting users within Europe and across the Atlantic. For years, it was common to see mail addresses of the form:
localnode!usanode!mcvax!euronode!user
where *mcvax* - named for the VAX at the Mathematical Centre (the name of CWI's parent foundation) played a central switching role for most of the inter- and intra-European UNIX network traffic.

During the past years, much has changed in the networking world, resulting in changes to our infrastructure and our role within the international networking community. First, *mcvax* has been replaced by *mcsun*, a Sun Microsystems SUN-4/280. Second, most addresses now consist of a user name and a destination host, with all of the intermediate hosts (and CWI's continued role) being supplied silently. Third, many of the responsibilities associated with maintaining networks have become more formalized, with the result that the management of UNIX networking has been transferred to broadly-based user's groups. For example, within The Netherlands, the foundation NLnet now manages most aspects of the Dutch national network of UNIX users, and EUnet manages many aspects of European-wide UNIX networking. While CWI continues to play an important role within the European networking community (evidenced by the fact that CWI remains the central node in a star-configured European UNIX network and that we continue to offer the technical support for Dutch and European UNIX network infrastructures), we also have been active in pursuing new directions for our networking work.

One major new direction for us has been a broadening of our networking focus. As is clear from the above paragraph, much of our emphasis has been on UNIX-based networking. While this area is still an active concern of ours, a group of people within the department of Computer Systems and Telematics (CST) has also been monitoring networking changes nationally and internationally. One clear development within Europe are plans for the development of networks based on the International Standard Organization's Open System Interconnection model (the so-called ISO/OSI or OSI protocols). The development of OSI-based networks is something that we watch with interest, although it falls outside of our own area of detailed expertise. Another development across Europe has been the proliferation in networks that, like the UNIX-based EUnet, are based on the use of the Internet Protocol (IP) family. IP has long been a North American *de facto* standard and has been very popular with computing equipment vendors, making it widely available on a wide range of scientific computers. IP networking in Europe is not new: some networks, such as NORDUnet and EUnet, have been actively using IP for several years. Newer networks, such as the IBM-sponsored European academic supercomputing initiative network EASInet, indicate that IP networking is still gaining in importance.



Main European International Leased Lines using IP.
Copyright CERN Geneva

International IP Leased Line (running IP over part or all of the indicated bandwidth)
National connection (not all are represented)
National IP Net.
International IP Net.
Multi-International IP Node

RIPE: Réseaux IP Européens

Given our long term interest in developing EUnet as cost-effective mean of interconnecting various UNIX systems, we participated (with several other groups) in the formation of a technical coordination group to facilitate cooperation among various IP networks. The name given to the IP-networking coordinating group is *Réseaux IP Européens*, or RIPE. The goal of this group is to achieve maximum interconnectivity between the various IP-based wide area networks in Europe. RIPE hopes to play an important technical role in formalizing the existing *ad hoc* agreements that exist among regional, national, and international networks with regard to interconnection strategies, packet routing, access restrictions and responsibilities, and (naturally) funding. We feel that RIPE can play an important role in reducing the costs of interconnecting networks who already use the IP protocols, while also providing a possible migration path to other protocols in the future.

As a technical forum for managers of IP-based networks, RIPE hopes to fill a void within European IP networking; since there was no formal pan-European coordination, there was no method of coordinating routing activity and network management or troubleshooting. RIPE was consciously set up in parallel to other European umbrella organizations to ensure broad participation from academia as well as industry, and from organizations operating on a national level as well as international and even local organizations. The emphasis in RIPE is clearly technical and operational. It is important that RIPE work remains focused because of the limited human and material resources available to scientific networking. In particular RIPE has been set up outside the international RARE organization, although close informal ties between RARE and RIPE are guaranteed because of a high degree of cross membership in the two groups. Since RARE has formally acknowledged the usefulness of IP networking for its particular target group, talks between RARE and RIPE about formal relations have started. RARE can play an important coordinating role in the planning of a pan-European multi-protocol networking infrastructure for the scientific community. RIPE can give important input on the greatly needed IP parts of this infrastructure.

An essential aspect of the RIPE construction is that individual networks retain total autonomy as far as their choice of protocols and their access policies are concerned. RIPE provides a mechanism to coordinate activity at an international level with regard to general services and new networking developments (such as new protocols or problem fixes), as well as providing a warning mechanism in the case of network service disruptions. It is also an information-sharing mechanism for internal and external groups who wish to discuss networking issues.

At present, the major groups involved in RIPE along with CWI/EUnet are NIKHEF/CERN/HEPnet, the Scandinavian NORDUnet, the Swiss SWITCH network, the French INRIA group and the Italian academic network GARR. The accompanying figure illustrates the general structure of the present RIPE backbone. The RIPE network database already contains entries for over 500 networks throughout Europe. Many of these networks have connected-status access to the US Internet, making the RIPE effective community quite extensive. Among the initial RIPE agreements have been the coordination of routing policies among these networks and the development of backup routing for participating networks.

Main activities

A first area of concern for RIPE has been the coordination of shared use of a common backbone segment that will multiplex a number of high-speed connections between several central IP nodes within Europe. By pooling resources and coordinating services, RIPE allows a number of networks that already communicate with each other to work more effectively and efficiently. As a result of this sharing, a number of separate high-speed networking links between Amsterdam and CERN will be combined into a single, higher-speed link. This combined link will provide shared services for EUnet, HEPnet, EASInet and EARN (the latter being a major academic network among European university computer centres).

RIPE has formed four task forces to address some of the most immediate networking problems within its community:

1. Connectivity and routing
This group is interested in providing an efficient and reliable means of passing information among networks at the packet level and access information at the user level. Given the fact that The Netherlands alone has close to 100 IP networks registered across its universities, colleges, government installations and public and private research centres, this is not a trivial task. The rate at which new IP networks are being created is increasing, meaning that systematic methods for updating routing tables and providing (transparent) network and host name recognition is increasing in importance.

2. Network management and operations
This group is interested in maintaining stable network services and providing a mechanism for preventing unwanted or unauthorized access. It is also interested in providing members with usage information, so that network growth can be coordinated. The deceptively simple tasks of establishing networks often hides the very time-consuming tasks required to stabilize operations, especially across a broad range of not-always-compatible national and internal transmission service providers.

3. Domain name service and management
This group is primarily interested in establishing effective logical connections within the IP community and across protocol gateways. The domain name service is a global service based on distributed databases; it provides name-to-address translation (and vice-versa) across the various IP networks that can be considered to comprise a world-wide internet. The domain name service not only provides address translation, however: it also guides mail delivery/routing and therefore bears direct relation to routing in general. Managing the domain name service encompasses management of the domain name space and coordination with the X.400 community for gatewaying purposes.

4. Formal coordination
The fourth group is interested in providing a common access point for constituent networks so that a number of broader-scale issues can be discussed. These issues include the development of common policies with the various public and private service carriers (such as the European PTT's and the North American private communications companies), as well as the development of migration paths to new network protocols and standards. This group also provides a mechanism for external contact with RIPE members on issues of shared access and policy issues.

On behalf of EUnet and given our expertise in the first three areas, CWI plays an active role in the connectivity, network management and domain service technical groups.

CWI's interest
EUnet and CWI have a number of interests that have made RIPE participation attractive. First, RIPE as an organization can serve as a responsible force in coordinating activity among groups that already have a great deal of common interconnections. This means that CWI's researchers have access to a broad range of colleagues at many different educational, commercial, and research communities. Second, RIPE provides a forum in which new network management strategies can be tested and evaluated. This is especially important as migrations to universal protocols (some based on ISO/OSI definitions and others perhaps newly developed to meet specific needs encountered by RIPE members) begins to become feasible. Third, the flat structure of its organization means that RIPE can provide a responsive and highly efficient forum. Finally, we see RIPE as a logical extension of work already being done by CWI within the context of EUnet, and thus provides a means of our sharing our expertise with other interested parties.

PROJECTS

The following information is given for each project: a short description, the start-up year, research staff (project leader in italic) and cooperating institutions.

Internet/ISO protocol development

As the desire to communicate electronically among researchers within Europe (and worldwide) increases, it becomes important to expand the reach of individual computer networks. The Internet/ISO protocol development project is aimed at realizing this goal by investigating means of implementing protocols and protocol converters which will allow user-level network traffic to be routed between networks that support varying protocols. In particular, this project is aimed at providing protocol translation facilities between Internet protocols (primarily RFC-822) and ISO protocols (X.400) for message and mail traffic. (1989)

University College London, Nottingham Univ., RARE, other network members (EARN, EUnet, Janet).

P. Beertema, D.F. Karrenberg

Network-based task submission and performance monitoring

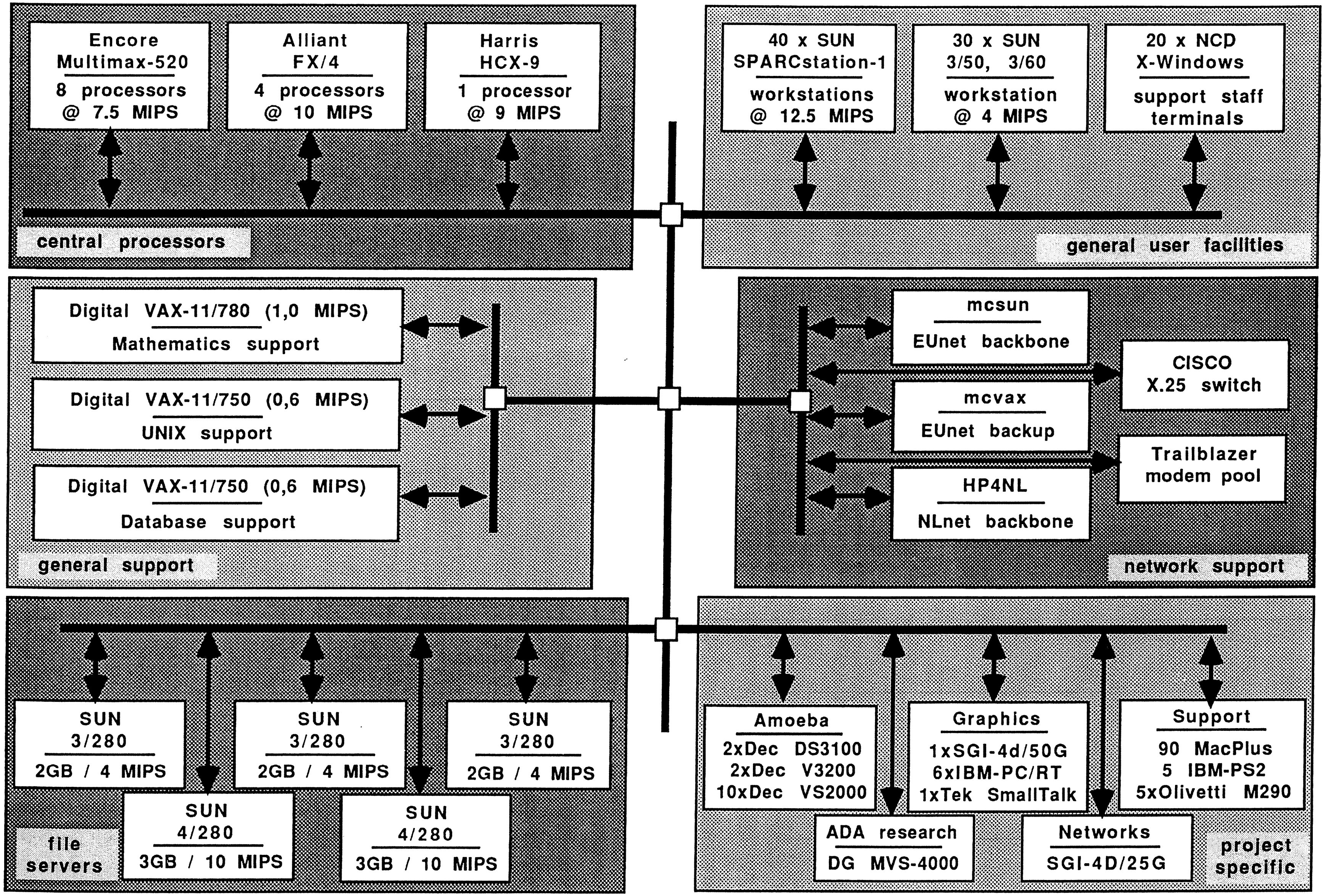
The Networked Execution Server (NES) project is aimed at studying the use of a collection of compute server computers in a coordinated manner for remote execution (and parallel execution) of user programs in a workstation-oriented environment. In particular, we are interested in studying three aspects of this problem:

- Task Allocation and Resource Sharing within a network of compute server, compute client, and monitoring computers;
- Communications Structures for supporting high-speed communication, possibly including hardware interconnection issues and hardware/software protocol issues;
- User Interfaces for graphically defining execution streams as collections of user-defined and system-defined components located across the network, and for displaying the status of both the execution stream and the system as a whole in user- and system-defined forms.

The work is constrained by the desire to use existing communications equipment where practical (i.e., Ethernet), although not necessarily existing network communication protocols. (1989)

D.C.A. Bulterman

CWI Computing Equipment Resources (as of December 31, 1989)



This chapter summarizes the various large-scale projects in which CWI participates. Whilst there is nothing new about cross-border contacts among scientists, recent years have seen a boom in national and international cooperation. The list of such programmes involving CWI grows apace, year after year.

The following data are given for each project:
- title,
- period,
- cooperation with other institutes,
- special role of CWI (if any),
- CWI project leader(s).

European Programmes

ESPRIT I

METEOR (432): An integrated formal approach to industrial software development
October 1984 - October 1989
Philips (PRLB, PRLE), CGE Paris, AT&T/Philips, COPS, TXT, Univ. Passau
J.A. Bergstra

GIPE (348): Generation of Interactive Programming Environments
November 1984 - November 1989
INRIA, SEMA, BSO
P. Klint

415: Parallel architectures and languages for AIP - a VLSI directed approach
November 1984 - November 1989
Subcontractor of Philips
J.W. de Bakker

ESPRIT II

GIPE II (2177): Generation of Interactive Programming Environments II
January 1989 - January 1992
Sema Metra (coordinator), INRIA, The Netherlands PTT, Planet, GIPSI, Bull
P. Klint

TROPICS (2427): Transparent Object-oriented Parallel Information Computing System
December 1988 - March 1990
Nixdorf, Olivetti, Thomson, Philips, CAP Sogeti
Associated contractor of Philips
M.L. Kersten

ATMOSPHERE (2565): Advanced Systems Engineering Environments
March 1989 - March 1991
Siemens, Bull, Société Française de Génie Logiciel, ESF Association, GEC-Marconi, Nixdorf, Philips
Associated contractor of Philips
J.C.M. Baeten

ESPRIT Basic Research Action (BRA)

CONCUR (3006): Theory of Concurrency: Unification and Extension
September 1989 - September 1991
Univ. Amsterdam, Univ. Edinburgh, Univ. Sussex, Univ. Oxford, Swedish Institute of Computer Science, INRIA
Coordinator
J.C.M. Baeten

INTEGRATION (3020): Integrating the Foundations of Functional, Logic and Object-oriented Programming
July 1989 - January 1992
CAIMENS, Philips, Università di Pisa, Centro de Inteligencia Artificial, Imperial College
Coordinator
J.W. de Bakker

SEMAGRAPH (3074): Semantics and Pragmatics of Generalized Graph Rewriting
July 1989 - January 1992
Univ. East Anglia (coordinator), CNRS, Imperial College, Univ. Nijmegen, ICL
J.W. Klop

RACE

RIPE (1040): RACE Integrity Primitives Evaluation
November 1988 - January 1991
Other consortium members: Siemens AG, Philips Usfa BV, The Netherlands PTT

Research, Universities of Leuven and Aarhus
Prime contractor
D. Chaum

SPECS (1046): Specification and Programming Environment for Communication Software
January 1988 - January 1993
Subcontractor of The Netherlands PTT Research
F.W. Vaandrager

Other programmes

ESA project HERMES: Convergence acceleration and accuracy improvement of a geometric adaptive multigrid method for finite volume Euler and Navier-Stokes flow computations
La Société Avions Marcel Dassault, Breguet Aviation
July 1987 - January 1991
P.W. Hemker

CODEST project EUROMATH: The integrated database and communications system for European mathematicians
January 1988 - June 1989
DDC, EMT, NIHE
L.G.L.T. Meertens

OSF (Open Software Foundation)

S.J. Mullender

National Programmes

SPIN (Stimulation Project Team Computer Science)

PRISMA: Parallel Inference and Storage Machine
October 1986 - October 1990
Philips (main contractor), Universities of Twente, Utrecht and Amsterdam
M.L. Kersten/P.J.F. Lucas

FLAIR: Flexible automation
January 1987 - January 1991
Univ. Twente
P.J.W. ten Hagen

PARTOOL: A parallel processing development environment
January 1989 - January 1993
TNO (coordinator), Philips, Techn. Univ. Delft, Univ. Utrecht
J.K. Lenstra

SION (Netherlands Foundation for Computer Science)

National Concurrency Project
May 1984 - February 1989
Techn. Univ. Eindhoven, Univ. Leiden
J.W. de Bakker

Mathematical morphology in hierarchical graph representations of images
TNO, Univ. Amsterdam
H.J.A.M. Heijmans

NFI (National Facility Computer Science)

Design methods for decision support systems
September 1985 - September 1989
Technical Univ. Delft, Technical Univ. Eindhoven, Erasmus Univ. Rotterdam
J.K. Lenstra

Cryptography and computer security
September 1984 - September 1990
D. Chaum

Research and Education in Concurrent Systems (REX)
January 1988 - January 1993
Technical Univ. Eindhoven, Univ. Leiden
J.W. de Bakker

Transformational programming
January 1988 - January 1993
Univ. Nijmegen, Univ. Utrecht
L.G.L.T. Meertens

Intelligent CAD systems
October 1986 - January 1993
TNO/IBBC, Univ. Amsterdam
P.J.W. ten Hagen

Formal methods for the description of information systems and their analysis
September 1989 - September 1993

Universities of Eindhoven, Leiden, Limburg and Twente
M.L. Kersten

STW (Foundation for the Technical Sciences)

Two-dimensional time-dependent Boussinesq model
August 1988 - July 1990
P.J. van der Houwen

Statistical analysis of debugging and error counting models in software reliability
March 1989 - March 1992
Univ. Utrecht
K.O. Dzhaparidze

Prediction and control problems for the Dutch freeway control and signalling system
January 1986 - January 1990
Univ. Twente
J.H. van Schuppen

Overload control for communication systems
February 1986 - February 1990
Philips Telecommunication, Univ. Twente
J.H. van Schuppen

Adaptive grid techniques for evolutionary partial differential equations
September 1987 - September 1990
Shell
J.G. Verwer

New architecture for interactive raster graphics on the basis of VLSI
April 1987 - April 1990
Univ. Twente
P.J.W. ten Hagen

IOP (Innovative Research Programmes)

IC-Technology: numerical methods for semiconductor device modelling
October 1987 - February 1992
FOM, Technical Univ. Delft, Philips CAD-Centre Eindhoven
P.W. Hemker

SPI (Special Programme Computer Science)

Algorithms and Complexity
January 1989 - January 1991

P.M.B. Vitányi

Shell fellowships

Mixed integer programming models for distribution and production planning
November 1987 - November 1989
Shell Research
J.K. Lenstra

Inverse scattering and image processing of seismic data
January 1988 - January 1990
Shell Research
J.H. van Schuppen

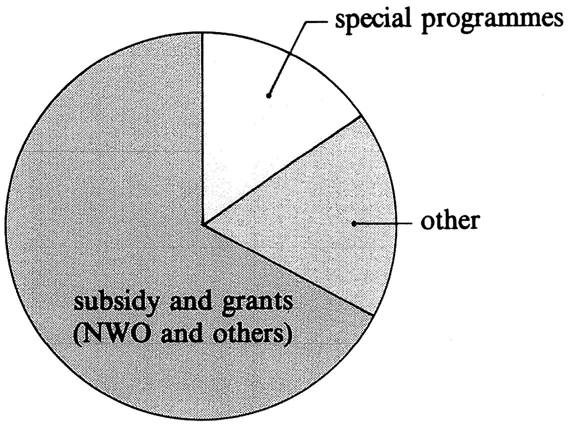
Financial Data, Personnel

FINANCES 1989

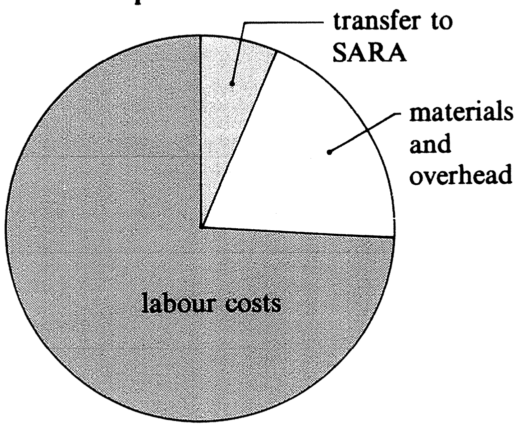
In 1989, SMC spent Dfl. 20.43 million, of which about Dfl. 1.68 million was allocated to research by the national working parties and Dfl. 18.75 million to CWI.
The expenses were covered by a subsidy from NWO (Dfl. 14.39 million), other subsidies and grants (Dfl. 0.46 million), from the European Community for its ESPRIT, CODEST and RACE projects (Dfl. 2.20 million), and from national programmes (Dfl. 1.11 million). Finally, an amount of about Dfl. 2.87 million was obtained as revenues out of third-party-services and other sources.
During 1989 CWI also employed 23 researchers in externally financed positions, for example by STW and industry. These are not included in the adjacent financial summary.

	national working parties	*CWI*	*SMC*
	× *Dfl.* 1000		
INCOME			
subsidy and grants			
- NWO	1725	12665	14390
- other	-	460	460
national programmes			
- INSP	-	545	545
- SPIN	-	540	540
- IOP	-	28	28
international programmes			
- ESPRIT	-	1622	1622
- CODEST	-	203	203
- RACE	-	372	372
other			
- proceeds from services and courses	-	1185	1185
- sales of publications	-	154	154
- network services	-	595	595
- miscellaneous income	-	932	932
total income	1725	19301	21026
EXPENSES			
labour costs	1610	13878	15488
materials and overhead	68	3697	3765
transfer to SARA	-	1177	1177
total expenses	1678	18752	20430

income CWI

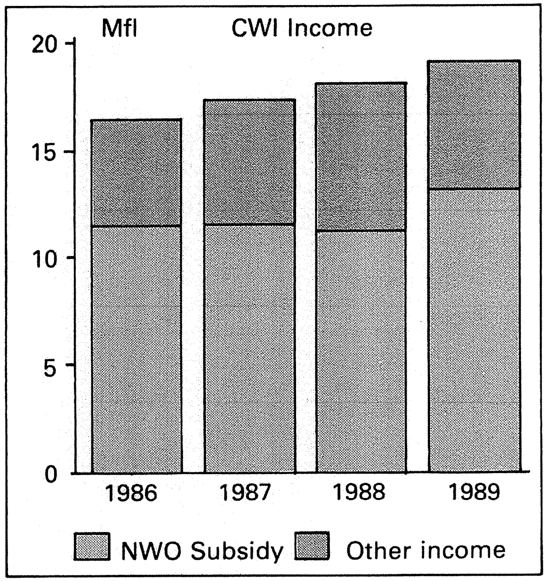


expenses CWI

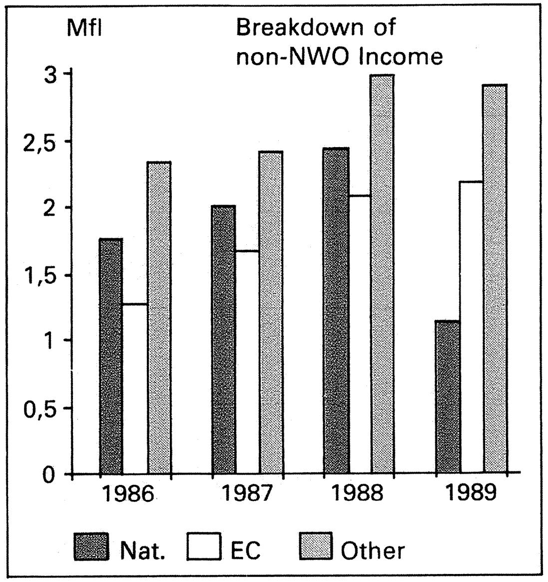


CWI Budget Computer Equipment		
X *Dfl.* 1000		
NWO	550	
IAS	1750	
total income (= expenses)	2300	

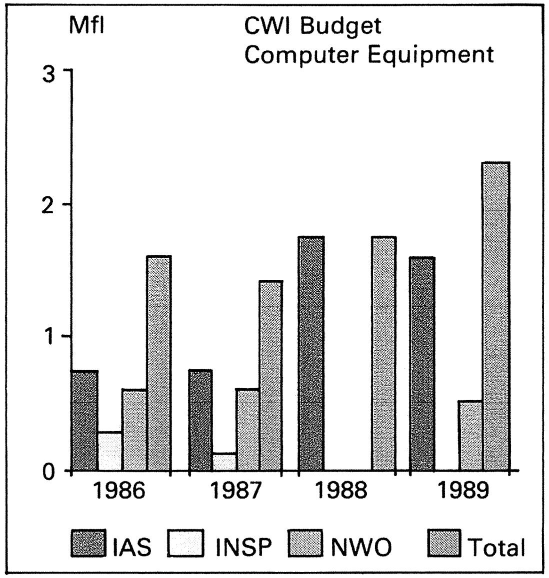
FINANCES 1986-1989



NWO grant as proportion of total operating costs for 1986-1989. After a steady decrease this proportion showed in 1989 some growth. This was due to a drop in income caused by the termination of the INSP stimulation programme, which was partly compensated by NWO.

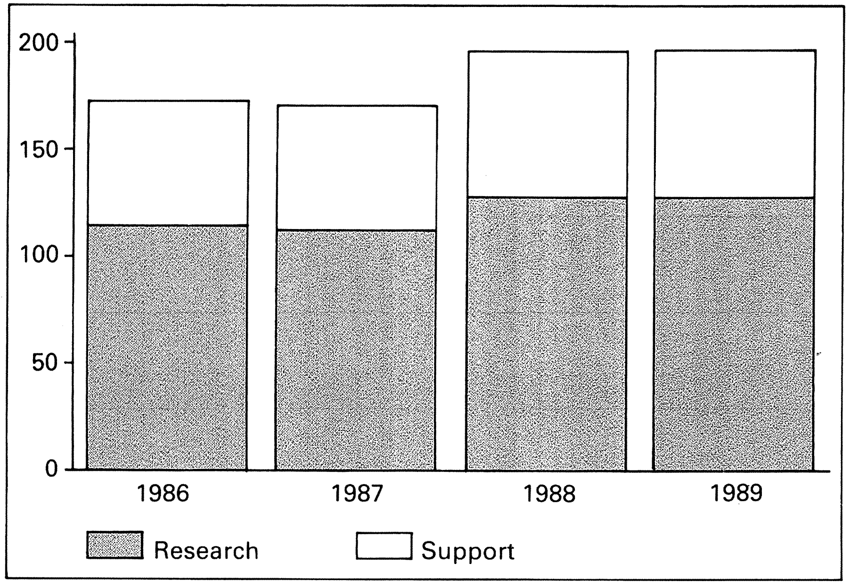


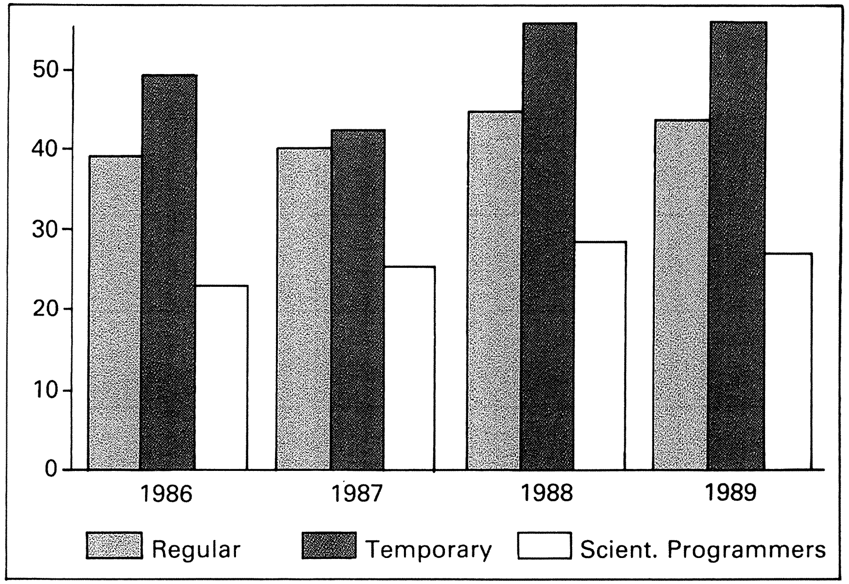
Contrary to the decrease in income from national programmes, income from EC programmes in 1989 still shows some growth. CWI participation in EC programmes has now come close to the maximum attainable.



Mainly due to the grant from the IAS government support scheme for scientific equipment, CWI's computer equipment situation further improved, although funding is still inadequate by international standards.

PERSONNEL 1986-1989





The size of the personnel force is expressed in full-time equivalents, averaged over the year in question. Not included are externally financed positions (e.g. from STW and industry). For the years 1986-1989 these amounted to 10, 17, 21 and 23 respectively. To the right the breakdown of the research personnel is shown.

Foreign Visitors

Analysis, Algebra and Geometry

J-P. Antoine (Belgium)
M. Ya Antonovsky (Austria)
R. Askey (USA)
P.J. Cameron (UK)
V. Capasso (Italy)
L. Collatz (FRG)
K. Dietz (FRG)
S.G. Gindikin (USSR)
M. Lindgren (Sweden)
V.S. Mihailovitch (USSR)
R. Murenzi (Belgium)
H. Petzold (FRG)
O. Ragnisco (Italy)
B. Randell (UK)
C. Ronse (Belgium)
S.S. Sazhin (UK)
E. Scholz (FRG)
G. Stroth (FRG)
J.A. Thas (Belgium)
H.R. Thieme (USA)
J.G. Timmesfeld (FRG)
L. Vincent (France)
A.M. Vinogradov (USSR)
V.S. Vladimirov (USSR)
H-O. Walther (FRG)
Zelazko (Poland)
D. Zvirenaite (USSR)

Operations Research, Statistics, and System Theory

K. Aardal (Belgium)
A.C. Antoulas (USA)
J.E. Besag (UK/USA)
C. Bezuidenhout (UK/USA)
R.M. Burton (USA)
M.A. Cameron (Australia)
R. Coleman (UK)
L. Cruz (Switzerland)
H. Daduna (FRG)
P.J. Diggle (UK)
N.I. Fisher (Australia)
P. Franken (GDR)
P.A. Fuhrmann (Israel)
N. Gamkrelidze (USSR)
E.G. Golshtein (USSR)
G.R. Grimmett (UK)
H.J. Gundersen (Denmark)
R.Z. Has'minskii (USSR)
B. Jacubczyk (Poland)
C. Jennison (UK)
E.B. Jensen (Denmark)
I.M. Johnstone (USA)
F.P. Kelly (UK)
J.H.B. Kemperman (USA)
H. Kesten (USA)
D. König (GDR)
T. Konstantopoulos (USA)
P.S. Krishnaprasad (USA)
P.J. Kuehn (FRG)
E.V. Levner (USSR)
H. Levy (Israel)
T. Mattfeldt (FRG)
J. Mecke (GDR)
R.E. Miles (Canada)
J. Moller (Denmark)
F. Montes (Spain)
K. Murota (Japan)
Ph. Nain (France)
D.Q. Neiman (USA)
C.M. Newman (USA)
N. Nikoukhah (France)
A.A. Novikov (USSR)
T.J. Ott (USA)
L. Pandolfi (Italy)
B. Reed (Canada)
B. Ripley (UK)
C. Ronse (Belgium)
H. Rootzén (Sweden)
S.M. Ross (USA)
L. Russo (Italy)
M. Schmitt (France)
M. Smorodinsky (Israel)
J. Stoyanov (Bulgaria)
C.C. Taylor (UK)
R.A. Vitale (USA)
W. Weil (FRG)
J.A. Wieacker (FRG)
W. Wolf (GDR)
A.M. Yaglom (USSR)
U. Yechiali (Israel)

Numerical Mathematics

U. Ascher (Canada)
A. Bellen (Italy)
H. Brunner (Canada)
K. Burrage (UK)
M.G. Cox (UK)
J.A. Desideri (France)
C.C. Douglas (USA)
H.C. Fischer (FRG)
G.S. Hodgson (UK)
B Karasözen (Turkey)
R. Kelch (FRG)
R. Klatte (FRG)
J.J. Lauture (Belgium)
B. van Leer (USA)
T. Mitsui (Japan)
S. Scholz (GDR)
G. Schumacher (FRG)
Y. Shokin (USSR))
K. Strehmel (GDR)
P.J.L. Wallis (UK)
J. Wolff von Gudenberg (FRG)
M. Zennaro (Italy)
A. Zöllner (FRG)

Software Technology

R.-J. Back (Finland)
M. Broy (FRG)
L. Chaillan (France)
L. Cherkasova (USSR)
K. Clark (UK)
Ch. Codognet (France)
G. Cousineau (France)
G. David (Portugal)
M. Dauphin (France)
P. Degano (Italy)
J. van Eijck (UK)
M.C. Ferreira (Portugal)
X. Franc (France)
L. Fribourg (France)
K. Futatsugi (Japan)
J.C. Godskesen (Denmark)
M.C.B. Hennessy (UK)
C.A.R. Hoare (UK)

J.M. Jacquet (Belgium)
M. Josephs (UK)
R. Kennaway (UK)
V.E. Kotov (USSR)
E. Madeleine (France)
M. Mamede (Portugal)
F.G. McCabe (UK)
M. Michel (France)
L. Monteiro (Portugal)
C. Palamidessi (Italy)
M. Phalippou (France)
S. Pickin (France)
G.D. Plotkin (UK)
A. Porto (Portugal)
D.S. Scott (USA)
R.K. Shyamasundar (India)
M.R. Sleep (UK)
S. Smolka (USA)
J.V. Tucker (UK)
K. Ueda (Japan)
D. Walker (UK)

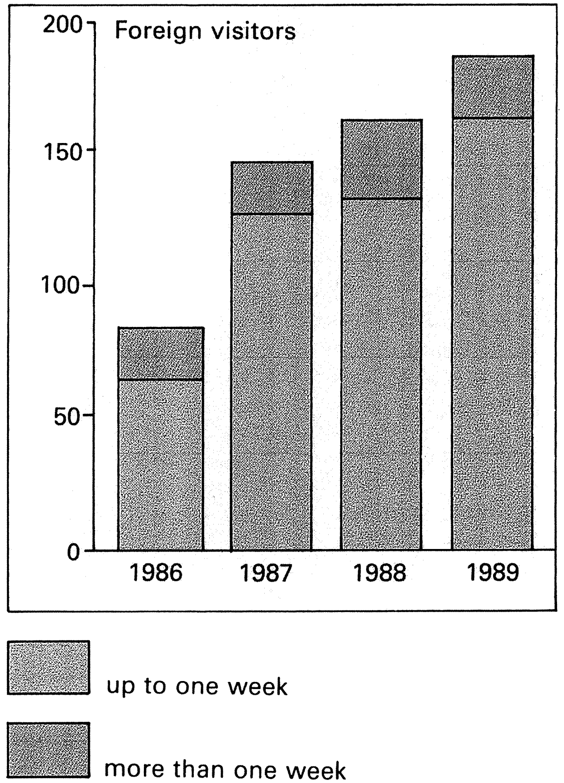
Algorithmics and Architecture

Ö. Babaõglu (Italy)
P. Baudelaire (France)
P. Bernadat (France)
T. Burkow (Sweden)
L. Cardelli (USA)
B. David (France)
Dukjoo Son (Korea)
M. Gereb-Graus (USA)
S.P. Ghosh (USA)
S.F. Hummel (USA)
G. Jones (UK)
H. Karloff (USA)
S. Karsenty (France)
G. Kissin (Canada)
L. Lamport (USA)
E. Murcier (Switzerland)
G. Nelson (USA)
O. Ninneman (FRG)
E. Paire (France)
M. Pocchiola (France)
J. Steiner (USA)
L. Stewart (USA)
J.L. Vicente (Spain)
Y. Viemont (France)
K. Yap (USA)

Interactive Systems

A. Bijl (UK)
U. Claussen (FRG))
D. Duce (UK)
J.L. Encarnacao (FRG)
J. Hübl (FRG)
W. Hübner (FRG)
M. Mehl (FRG)
E. Ostby (USA)
F. Yamamoto (Japan)

FOREIGN VISITORS 1986 - 1989



List of Publications

Department of Analysis, Algebra and Geometry

Report series

AM-R8901 J.K. SCHOLMA. *Lie-algebraic aspects of the classical nonrelativistic Calogero-Moser models.*
AM-R8902 G. ALBERTS. *Signific consultation; David van Dantzig's dream of a practical significs.*
AM-R8903 J.M.A.M. VAN NEERVEN. *Hahn-Banach type theorems for dual semigroups.*
AM-R8904 C. RONSE, H.J.A.M. HEIJMANS. *The algebraic basis of mathematical morphology, Part II: Openings and closings.*
AM-R8905 M. ZWAAN. *Dynamic MRI reconstruction as a moment problem, Part I. The beating heart: a problem formulation.*
AM-R8906 T.H. KOORNWINDER. *The addition formula for little q-Legendre polynomials and the SU(2) quantum group.*
AM-R8907 M. ZWAAN. *Dynamic MRI reconstruction as a moment problem, Part II. Riez bases in L^2-spaces of vector valued functions.*
AM-R8908 J.M.A.M. VAN NEERVEN. *Reflexivity, the dual Radon-Nikodym property, and continuity of adjoint semigroups.*
AM-R8909 A.M. COHEN, E.E. SHULT. *Affine polar spaces.*
AM-R8910 S.N.M. RUIJSENAARS. *Relativistic Toda systems.*
AM-R8911 N.M. TEMME. *Polynomial asymptotic estimates of Gegenbauer, Laguerre, and Jacobi polynomials.*
AM-R8912 O. DIEKMANN, J.A.P. HEESTERBEEK, J.A.J. METZ. *On the definition and the computation of the basic reproduction ratio R_0 in models for infectious diseases in heterogeneous populations.*
AM-R8913 H.J.A.M. HEIJMANS, A. TOET. *Morphological sampling.*
AM-R8914 M. ZWAAN. *Stability and error estimates for sinc interpolation.*
AM-R8915 H. INABA. *Threshold and stability for an age-structured epidemic model.*
AM-R8916 H. INABA. *Perturbation theory for dual semigroups and its applications to age-dependent population dynamics.*
AM-R8917 J.B.T.M. ROERDINK. *A Markov chain identity and monotonicity of the diffusion constants for a random walk in a heterogeneous environment.*
AM-R8918 M.J. COSTER. *Supercongruences.*
AM-R8919 A. GRABOSCH, H.J.A.M. HEIJMANS. *Production, development and maturation of red blood cells, A mathematical model.*
AM-R8920 J.A.J. METZ, O. DIEKMANN. *Exact finite dimensional representations of models for physiologically structured populations I: the abstract foundations of linear chain trickery.*
AM-R8921 J.B.T.M. ROERDINK, K.E. SHULER, G.F. LAWLER. *Diffusion in lattices with anisotropic scatterers.*
AM-R8922 A.B. OLDE DAALHUIS. *Asymptotic expansions of an integral containing a phase function with three saddle points.*
AM-R8923 M. ZWAAN. *Approximation of the solution to the moment problem in a Hilbert space.*
AM-R8924 J.B.T.M. ROERDINK. *Mathematical morphology on homogeneous spaces, Part I. The simply transitive case.*

Publications in Journals, proceedings, etc.

AM1 A.M. COHEN & B.N. COOPERSTEIN (1989). On the local characterization of finite metasymplectic spaces. *J. Algebra 125,* 348-366.
AM2 J. VAN BON, A.M. COHEN (1989). Linear groups and distance-transitive graphs. *European J. Combinatorics 10,* 399-411.
AM3 J. VAN BON, A.M. COHEN, H. CUYPERS (1989). Some graphs related to Held's simple group. *J. Algebra 123,* 6-26.
AM4 A.E. BROUWER, A.M. COHEN, A. NEUMAIER (1989). Distance-regular graphs. *Ergebnisse der Math., Vol.18,* Springer-Verlag, Berlin.
AM5 A.M. COHEN (1989). Computer algebra, wetenschap en techniek. *Proc. 25ste Nederlands Mathemathisch Congres.*

AM6 PH. CLÉMENT, O. DIEKMANN, M. GYLLENBERG, H.J.A.M. HEIJMANS, H.R. THIEME (1989). Perturbation theory for dual semigroups. Part III: nonlinear Lipschitz continuous perturbations in the sun-reflexive case. G. DA PRATO, M. IANNELLI (eds.). *Volterra Integro Differential Equations in Banach Spaces and Applications,* Longman, 67-89.

AM7 PH. CLÉMENT, O. DIEKMANN, M. GYLLENBERG, H.J.A.M. HEIJMANS, H.R. THIEME (1989). Perturbation theory for dual semigroups. Part IV: the intertwining formula and the canonical pairing. PH. CLÉMENT et al (eds.). *Trends in Semigroup Theory and Applications,* Marcel Dekker, New York, 95-116.

AM8 PH. CLÉMENT, O. DIEKMANN, M. GYLLENBERG, H.J.A.M. HEIJMANS, H.R. THIEME (1989). A Hille-Yosida type theorem for a class of weakly * continuous semigroups. *Semigroup Forum 38,* 157-178.

AM9 O. DIEKMANN, J.A.J. METZ (1989). Exploring linear chain trickery for physiologically structured populations, (afscheid Prof.dr. H.A. Lauwerier) TW in Beeld, 73-84. Reprinted as *CWI Quarterly 2,* (1989) 3-14.

AM10 O. DIEKMANN, M. GYLLENBERG, H.J.A.M. HEIJMANS (1989). When are two Co-semigroups related by a bounded perturbation? PH. CLÉMENT, S. INVERNIZZI, E. MITIDIERI, I.I. VRABIE (eds.). *Trends in Semigroup Theory and Applications,* Marcel Dekker, New York, 153-162.

AM11 O. DIEKMANN, J.A.J. METZ, M.W. SABELIS (1989). Reflections and calculations on a prey-predator-patch problem. *Acta Applicandae Mathematicae 14,* 23-35.

AM12 O. DIEKMANN (1989). On semigroups and populations. G. FUSCO, M. IANNELLI, L. SALVADORI (eds.). *Advanced Topics in the Theory of Dynamical Systems,* Academic Press, 125-135.

AM13 J. GRASMAN, J.B.T.M. ROERDINK (1989). Stochastic and chaotic relaxation oscillations. *J.Statist. Phys. 54,* 949-970.

AM14 M. HAZEWINKEL (1989). Nongaussian linear filtering, identification of linear systems, and the symplectic group. R. GIELERAK, W. KARWOVSKI (eds.). *Stochastic Methods in Physics,* World Scientific, 9, 325-337, also in: A. BLAQUIERE (ed.). *Modeling and Control of Systems,* Springer, 299-308, also in: H. NEUNZERT (ed.). *Proc. of the Second Workshop on the Road Vehicle System and Related Mathematics,* KAP-Teubner, 117-127.

AM15 M. HAZEWINKEL (1989). Extended review of J. Neukirch, Class field theory and K. Iwasawa, locla class field theory. *Bull AMS 21,* 95-101.

AM16 J.A.P. HEESTERBEEK, H.A.J.M. SCHELLINX (1989). On sums of remainders and almost perfect numbers. *CWI Quarterly 2,* 15-17.

AM17 J.A.P. HEESTERBEEK, J.C. ZADOKS (1989). Modelling pandemics of quarantine pests and diseases: problems and perspectives (in Chinese). *Plant Disease Management and Resistance Breeding Workshop FAO,* Lecture Notes, Beijing.

AM18 H.J.A.M. HEIJMANS (1989). Iteration of morphological transformations. *CWI Quarterly 2,* 19-36.

AM19 H.J.A.M. HEIJMANS, J.A.J. METZ (1989). Small parameters in structured population models and the Trotter-Kato theorem. *SIAM J. Math. Anal. Appl. 20,* 870-885.

AM20 M. HUŠEK, J. DE VRIES (1989). A note on compactifications of products of semigroups. *Semigroup Forum 38,* 85-89.

AM21 H. INABA (1989). Weak ergodicity of population evolution processes. *Math. Biosc. 96,* 195-219.

AM22 H.T. KOELINK, T.H. KOORNWINDER (1989). The Clebsch-Gordan coefficients for the quantum group SU(2) and q-Hahn polynomials. *Nederl. Akad. Wetensch. Proc. Ser. A 92,* 443-456.

AM23 T.H. KOORNWINDER (1989). Meixner-Pollaczek polynomials and the Heisenberg algebra. *J. Math. Phys. 30,* 767-769.

AM24 T.H. KOORNWINDER (1989). Representations of the twisted SU(2) quantum group and some q-hypergeometric orthogonal polynomials. *Nederl. Akad. Wetensch. Proc. Ser. A 92,* 97-117.

AM25 T.H. KOORNWINDER (1989). Continuous q-Legendre polynomials are spherical matrix elements of irreducible representations of the quantum SU(2) group. *CWI Quarterly 2,* 171-173.

AM26 M. KRETZSCHMAR (1989). A renewal equation with a birth-death process as a model for parasitic infections. *J. Math. Biol. 27,* 191-221.

AM27 M. KRETZSCHMAR (1989). Persistent solutions in a model for parasitic infections. *J. Math. Biol. 27,* 549-573.

AM28 A. GRANVILLE, J. VAN DE LUNE, H.J.J. TE RIELE (1989). Checking the Goldbach conjecture on a vector computer. R.A. MOLLIN (ed.). *Number Theory and Applications,* Kluwer Academic Publishers, 423-433.

AM29 J.B.T.M. ROERDINK (1989). The biennial life strategy in a random environment. Supplement. *J.Math. Biol. 27,* 309-319.

AM30 J.B.T.M. ROERDINK (1989). Products of random matrices or 'Why do biennials live longer than two years?' *CWI Quarterly 2,* 37-44.

AM31 H.N.M. ROOZEN (1989). A short introduction to exit problems. *CWI Quarterly 2,* 45-65.

AM32 H.N.M. ROOZEN (1989). Stochastic stability of the loaded stiff rod. *Journal of Engineering Mathematics 23,* 357-376.

AM33 H.N.M. ROOZEN (1989). An asymptotic solution to a two-dimenisonal exit problem arising in population dynamics. *SIAM J. Appl. Math. Vol. 49, No. 6,* 1793-1810.

AM34 G.C.M. RUITENBURG (1989). Invariant ideals of polynomial algebras with multiplicity free group action. *Compositio Mathematica 71,* 181-227.

AM35 S.N.M. RUIJSENAARS (1989). Dirac quantization, Virasoro and Kac-Moody algebras, and boson-fermion correspondence in 2N dimensions. B. SIMON, A. TRUMAN, I.M. DACIES (eds.). *Proc. 1988 Swansea International Congress on Mathematical Physics,* IOP Publishing, Bristol, 474-482.

AM36 S.N.M. RUIJSENAARS (1989). Index formulas for generalized Wiener-Hopf operators and boson-fermion correspondence in 2N dimensions. *Comm. Math. Phys. 124,* 553-593.

AM37 N.M. TEMME (1989). Asymptotic expansion of a special integral. *CWI Quarterly, 2,* 67-72.

AM38 N.M. TEMME (1989). On the computation of special functions by using asymptotic expansions. W.F. AMES, C. BREZINSKI (eds.). *IMACS Annals of Computing and Applied Mathematics: IMACS Transactions on Scientific Computing,* 12th IMACS World Congress, Paris, Vol. 1: Numerical and Applied Mathematics.

AM39 H.R. THIEME, J.A.P. HEESTERBEEK (1989). How to estimate the efficacy of periodic control of an infectious plant disease. *Math. Biosc. 93,* 15-29.

AM40 J. DE VRIES (1989). Invariant measures in abstract topological dynamics. *Proc. Conf. Topology & Measure V* (Binz, DDR, 1987), Wissenschaftliche Beiträge der Ernst-Moritz-Arndt Univ., Greifswald, 1988, 175-183.

AM41 M. ZWAAN (1989). A Radon transform on circles through the origin. *CWI Quarterly, 2,* 73-79.

Other publications

AM42 G. ALBERTS, H.J.M. BOS, J. NUIS (red.) (1989). Om de Wiskunde; Stimulansen voor Toepassingsgerichte Wiskunde rond 1946, CWI, Amsterdam.

AM43 W. VAN ASSCHE, T.H. KOORNWINDER (1989). *Asymptotic Behaviour for Wall Polynomials and the Addition Formula for Little q-Legendre Polynomials,* preprint.

AM44 A.M. COHEN, B. LISSER, B. RUITENBURG, B. DE SMIT, R. SOMMELING (1989). *LiE*, software , manual and tutorial.

AM45 M. HAZEWINKEL (1989). *Encyclopaedia of Mathematics, Vol. 3,* KAP, January.

AM46 M. HAZEWINKEL (1989). *Encyclopaedia of Mathematics, Vol. 4,* KAP, August.

AM47 T.H. KOORNWINDER (1989). *Orthogonal Polynomials in Connection with Quantum Groups,* preprint.

AM48 T.H. KOORNWINDER, R.F. SWARTTOUW (1989). *On q-*

Analogues of the Fourier and Hankel Transforms. Report 89-89, TU Delft, Fac. Techn. Wiskunde en Informatica.

Department of Operations Research, Statistics, and System Theory

Report series

BS-R8901 R.J. Chitashvili. *On the nonexistence of a strong solution in the boundary problem for a sticky Brownian motion.*
BS-R8902 P.R. de Waal, N.M. van Dijk. *Monotonicity of performance measures in a processor sharing queue.*
BS-R8903 M. Kuijper, J.M. Schumacher. *Realization of autoregressive equations in pencil and descriptor form.*
BS-R8904 E.V. Khmaladze. *Theory of goodness of fit tests and scanning innovation martingales.*
BS-R8905 O.J. Boxma, J.A. Weststrate. *Waiting times in polling systems with Markovian server routing.*
BS-R8906 O.J. Boxma. *Workloads and waiting times in single-server systems with multiple customer classes.*
BS-R8907 J.L. van den Berg. *Simple approximations for second moment characteristics of the sojourn time in the M/G/1 processor sharing queue.*
BS-R8908 S. Kalikow. *Random Markov processes and uniform martingales.*
BS-R8909 E.L. Lawler, J.K. Lenstra, A.H.G. Rinnooy Kan, D.B. Shmoys. *Sequencing and scheduling: Algorithms and complexity.*
BS-R8910 J.W. van der Woude. *Almost disturbance decoupling by measurement feedback: a frequency domain analysis.*
BS-R8911 J.W. Cohen. *On a class of N-dimensional associated random walks and the hitting point identity.*
BS-R8912 S.A. van de Geer, L. Stougie. *On rates of convergence and asymptotic normality in the multiknapsack problem.*
BS-R8913 A. Marchetti Spaccamela, W.T. Rhee, L. Stougie, S.A. van de Geer. *Probabilistic analysis of the minimum weighted flowtime scheduling problem.*
BS-R8914 J.A. Hoogeveen, S.L. van de Velde. *Scheduling around a small common due date.*
BS-R8915 M.I. Dessouky, B.J. Lageweg, J.K. Lenstra, S.L. van de Velde. *Scheduling identical jobs on uniform parallel machines.*
BS-R8916 O.J. Boxma, H. Daduna. *Sojourn times in queueing networks.*
BS-R8917 J.W. Cohen. *Associated distributions in the analysis of two-dimensional random walks.*
BS-R8918 J.W. van der Woude. *On the structure at infinity of a structured system.*
BS-R8919 O.J. Boxma, J.W. Cohen. *The M/G/1 queue with permanent customers.*
BS-R8920 J.H. van Schuppen. *Stochastic realization problems.*
BS-R8921 A. Schrijver. *Applications of polyhedral combinatorics to multicommodity flows and compact surfaces.*
BS-R8922 A. Schrijver. *Short proofs on multicommodity flows and cuts.*
BS-R8923 A. Schrijver. *Disjoint homotopic paths and trees in a planar graph.*
BS-R8924 A. Schrijver. *On the uniqueness of kernels.*
BS-R8925 L. Lovász, A. Schrijver. *Cones of matrices and set-functions, and 0-1 optimization.*
BS-R8926 R.A. Moyeed, A.J. Baddeley. *Stochastic approximation of the MLE for a spatial point pattern.*
BS-R8927 L. de Haan, A.L.M. Dekkers. *Optimal choice of sample fraction in extreme-value estimation.*
BS-R8928 R. Has'minskii, I. Ibragimov. *On density estimation in the view of Kolmogorov's ideas in approximation theory.*
BS-R8929 J.W. van der Woude. *System identification of lossless layered media from input-output data.*
BS-R8930 A.M.H. Gerards. *On cutting planes and matrices.*

BS-R8931 W.P. GROENENDIJK, H. LEVY. *Performance analysis of transaction driven computer systems via queueing analysis of polling models.*

BS-R8932 O.J. BOXMA, H. LEVY, J.A. WESTSTRATE. *Optimization of polling systems.*

BS-R8933 K.O. DZHAPARIDZE, P.J.C. SPREIJ. *On SLLN for martingales with deterministic quadratic variation.*

BS-R8934 P.R. DE WAAL, N.M. VAN DIJK. *Interconnected networks of queues with randomized arrival and departure blocking.*

BS-N8901 J.N.T. SCHUIT. *Numerical solution of the Hamilton-Jacobi-Bellman equation for a freeway traffic flow control problem.*

Publications in Journals, proceedings, etc.

BS1 J.M. ANTHONISSE, J.K. LENSTRA, M.W.P. SAVELSBERGH (1989). De bochtige weg van idee naar implementatie. L. FORTUIN, P. VAN BEEK, J. WESSELS (eds.). *Op de snede tussen theorie en praktijk; een liber amicorum opgedragen aan Prof.Dr. J.F. Benders, ter gelegenheid van zijn pensionering,* Eindhoven, 153-164.

BS2 A.J. BADDELEY, J. MOLLER (1989). Nearest-neighbour Markov point processes and random sets. *International Statistical Review 57,* 89-121.

BS3 H.C.P. BERBEE (1989). Uniqueness of Gibbs measures and absorption probabilities. *Ann. Prob. 17,* 1416-1432.

BS4 H.C.P. BERBEE, W.TH.F. DEN HOLLANDER (1989). Tail triviality for sums of stationary random variables. *Ann. Prob. 17,* 1635-1645.

BS5 J.L. VAN DEN BERG (1989). Simple approximations for second moment characteristics of the sojourn time in the M/G/1 processor sharing queue. G. STIEGE, J.S. LIE (eds.). *Messung, Modellierung und Bewertung von Rechensystemen und Netze,* Springer, Berlin, 105-120.

BS6 J.L. VAN DEN BERG, O.J. BOXMA (1989). Sojourn times in feedback queues. D. PRESSMAR et al. (eds.). *Operations Research Proceedings 1988,* Springer, Berlin, 247-257.

BS7 J.L. VAN DEN BERG, O.J. BOXMA (1989). Sojourn times in feedback and processor sharing queues. M. BONATTI (ed.). *Proceedings ITC-12,* North-Holland, Amsterdam, 1467-1475.

BS8 J.L. VAN DEN BERG, O.J. BOXMA, W.P. GROENENDIJK (1989). Sojourn times in the M/G/1 queue with deterministic feedback. *Stochastic Models 5,* 115-129.

BS9 R.K. BOEL, J.H. VAN SCHUPPEN (1989). Distributed routing for load balancing. *Proc. IEEE 77,* 210-221.

BS10 O.J. BOXMA (1989). Workloads and waiting times in single-server systems with multiple customer classes. *Queueing Systems 5,* 185-214.

BS11 O.J. BOXMA, J.A. WESTSTRATE (1989). Waiting times in polling systems with Markovian server routing. G. STIEGE, J.S. LIE (eds.). *Messung, Modellierung und Bewertung von Rechensystemen und Netze,* Springer, Berlin, 89-104.

BS12 A.L.M. DEKKERS, L. DE HAAN (1989). On the estimation of the extreme-value index and large quantile estimation. *Annals of Statistics 19.*

BS13 A.L.M. DEKKERS, J.H.J. EINMAHL, L. DE HAAN (1989). A moment estimator for the index of an extreme-value distribution. *Annals of Statistics 19.*

BS14 F.A. VAN DER DUYN SCHOUTEN, T. RONNER. Calculation of the availability of a two-unit parallel system with cold standby: an illustration of the embedding technique. *Probability in the Engineering and Informational Sciences, 3,* 341-353.

BS15 A.M.H. GERARDS (1989). A short proof of Tutte's characterization of totally unimodular matrices. *Linear Algebra and Its Applications 114/115,* 207-212.

BS16 A.M.H. GERARDS (1989). A min-max relation for stable sets in graphs with no odd-K4. *Journal of Combinatorial Theory (Series B) 47,* 330-348.

BS17 W.P. GROENENDIJK (1989). Waiting-time approximations for cyclic-service systems with mixed service strategies. M.

BONATTI (ed.). *Proceedings ITC-12,* North-Holland, Amsterdam, 1434-1441.

BS18 HAN-FU CHEN, P.R. KUMAR, J.H. VAN SCHUPPEN (1989). On Kalman filtering for conditionally Gaussian systems with random matrices. *Systems and Control Letters 13,* 397-404.

BS19 C.A.J. HURKENS, A. SCHRIJVER (1989). On the size of systems of sets every t of which have an SDR, with an application to the worst-case ratio of heuristics for packing problems. *SIAM Journal on Discrete Mathematics 2,* 68-72.

BS20 G.A.P. KINDERVATER, J.K. LENSTRA, D.B. SHMOYS (1989). The parallel complexity of TSP heuristics. *J. Algorithms 10,* 249-270.

BS21 G.A.P. KINDERVATER, J.K. LENSTRA, A.H.G. RINNOOY KAN (1989). Perspectives on parallel computing. *Operations Research 37,* 985-990.

BS22 G.A.P. KINDERVATER, J.K. LENSTRA, M.W.P. SAVELSBERGH (1989). Parallel local search for the time-constrained traveling salesman problem. J.K. LENSTRA, H.C. TIJMS, A. VOLGENANT (eds.). *Twenty-Five Years of Operations Research in the Netherlands; Papers Dedicated to Gijs de Leve,* CWI Tract 70, CWI, Amsterdam, 61-75.

BS23 M. KUIJPER, J.M. SCHUMACHER (1989). Transformations under external equivalence for descriptor representations. *Preprints IFAC Workshop on System Structure and Control: State-space and Polynomial Methods,* Prague, September 25-27, 1989, 17-20.

BS24 L. LOVÁSZ, M. SAKS, A. SCHRIJVER (1989). Orthogonal representations and connectivity of graphs. *Linear Algebra and Its Applications 114,* 439-454.

BS25 A. SCHRIJVER (1989). Distances and cuts in planar graphs. *Journal of Combinatorial Theory (B) 46,* 46-57.

BS26 A. SCHRIJVER (1989). Homotopy and crossings of systems of curves on a surface. *Linear Algebra and Its Applications 114,* 157-167.

BS27 A. SCHRIJVER (1989). The Klein bottle and multicommodity flows. *Combinatorica 9,* 373-382.

BS28 A. SCHRIJVER (1989). On the strong connectivity problem. J.K. LENSTRA, H.C. TIJMS, A. VOLGENANT (eds). *Twenty-Five Years of Operations Research in the Netherlands; Papers Dedicated to Gijs de Leve,* CWI Tract 70, CWI, Amsterdam, 122-126.

BS29 J.M. SCHUMACHER (1989). Linear system representations. H. NIJMEIJER, J.M. SCHUMACHER (eds.). *Three Decades of Mathematical System Theory. A Collection of Surveys at the Occasion of the 50th Birthday of Jan C. Willems,* Lecture Notes in Control and Information Science 135, Springer-Verlag, Berlin, 382-408.

BS30 J.M. SCHUMACHER (1989). Discrete events: perspectives from system theory. *CWI Quarterly 2,* 131-146

BS31 J.M. SCHUMACHER (1989). Analytic representations of structural dynamics. *Proc. ESTEC Workshop on Modal Representations of Flexible Structures by Continuum Methods,* Noordwijk, June 15-16, 1989, 143-156.

BS32 J.H. VAN SCHUPPEN (1989). Stochastic realization problems. H. NIJMEIJER, J.M. SCHUMACHER (eds.). *Three Decades of Mathematical System Theory. A Collection of Surveys at the Occasion of the 50th Birthday of Jan C. Willems,* Lecture Notes in Control and Information Sciences 135, Springer-Verlag, Berlin, 1989, 480-523.

BS33 S.A. SMULDERS (1989). Application of stochastic control concepts to a freeway traffic control problem. *CCCT'89, Control, Computers, Communication in Transportation,* Paris, 19-21 September, 295-301.

BS34 P.R DE WAAL (1989). An approximation method for a processor sharing queue with controlled arrivals and a waitbuffer. *Proc. of the 28th IEEE Conference on Decision and Control,* IEEE Press, Piscataway, 588-589.

Other publications

BS35 A.J. BADDELEY. Introduction to Markov spatial models. *Acta Stereologica,* to appear 1990.

BS36 J. VAN DEN BERG, E. KRANAKIS, D. KRIZANC. Computing Boolean Functions on Anonymous Networks. Preprint October 1989. Accepted for *ICALP 1990.*

BS37 A.L.M. DEKKERS, H. DAAN, D. DILLINGH, L. DE HAAN, R. HELMERS, J.W. VAN DER MADE, J. VAN MALDE, J.G. DE RONDE (1989). *Frequentielijnen Kust.* Tussentijds verslag 2, Nota GWIO 89.001, Rijkswaterstaat, Dienst Getijdewateren, 79 pp.

BS38 F.A. VAN DER DUYN SCHOUTEN, S.G. VANNESTE (1989). *Analysis and Computation of* (n,N)*-strategies for Maintenance of a Two Component System.* Research Memorandum 372, Dept. of Economics, Tilburg University.

BS39 K. DZHAPARIDZE, E. VALKEILA (1989). *On Large Deviation Probabilities for Maximum Likelihood Estimators. Submitted for Publication.*

BS40 R. HELMERS (1989). *Asymptotische statistiek en bootstrap methoden.* Syllabus, 79 pp.

BS41 J.K. LENSTRA, H.C. TIJMS, A. VOLGENANT (eds.) (1989). *Twenty-Five Years of Operations Research in the Netherlands; Papers Dedicated to Gijs de Leve,* CWI Tract 70, CWI, Amsterdam.

BS42 K. MUROTA, J.W. VAN DER WOUDE (1989). *Disturbance Decoupling and Structure at Infinity of Structured Descriptor Systems.* Report No. 89605-OR, Institut für Ökonometrie und Operations Research, Rheinische Friedrich-Wilhelms-Universität Bonn, Bonn.

BS43 H. NIJMEIJER, J.M. SCHUMACHER (eds.) (1989). *Three Decades of Mathematical System Theory. A Collection of Surveys at the Occasion of the 50th Birthday of Jan C. Willems,* Lecture Notes in Control and Information Science 135, Springer-Verlag, Berlin.

BS44 S.A. SMULDERS (1989). *Control of Freeway Traffic Flow.* Thesis, Universiteit Twente, Enschede.

BS45 W. DEN OTTER, O.J. BOXMA et al. (1989). *Oncogenesis by Mutations in Anti-oncogenes: a View.* Report Pathologisch Instituut, RU Utrecht.

BS46 W. DEN OTTER, O.J. BOXMA et al. (1989). *Hereditary Cancer and its Clinical Implications: a View.* Report Pathologisch Instituut, RU Utrecht.

Department of Numerical Mathematics

Report series

NM-R8901 W.H. HUNDSDORFER, J. SCHNEID. *An algebraic characterization of B-convergent Runge-Kutta methods.*

NM-R8902 B.P. SOMMEIJER. *Increasing the real stability boundary of explicit methods.*

NM-R8903 W. HOFFMANN, K. POTMA. *NUMVEC FORTRAN library manual. Chapter: Simultaneous linear equations.*

NM-R8904 J.G. BLOM, P.A. ZEGELING. *A moving-grid interface for systems of one-dimensional time-dependent partial differential equations.*

NM-R8905 P.M. DE ZEEUW. *Nonlinear multigrid applied to a 1D stationary semiconductor model.*

NM-R8906 P.J. VAN DER HOUWEN, B.P. SOMMEIJER. *Block Runge-Kutta methods on parallel computers.*

NM-R8907 J.G. VERWER, W.H. HUNDSDORFER, B.P. SOMMEIJER. *Convergence properties of the Runge-Kutta-Chebyshev method.*

NM-R8908 M.H. LALLEMAND, B. KOREN. *Iterative defect correction and multigrid accelerated explicit time stepping schemes for the steady Euler equations.*

NM-R8909 M. BERGMAN. *Testing of elementary functions in Ada.*

NM-R8910 E.D. DE GOEDE. *A computational model for three-dimensional shallow water flows on the ALLIANT FX/4.*

NM-R8911 W.H. HUNDSDORFER. *Convergence of one-leg multistep methods for stiff nonlinear initial value problems.*

NM-R8912 P.J. VAN DER HOUWEN, B.P. SOMMEIJER, W. COUZY. *Embedded diagonally implicit Runge-Kutta algorithms on parallel computers.*

NM-R8913 P.J. van der Houwen. *Block Runge-Kutta methods.*

NM-R8914 W.H. Hundsdorfer. *Local and global order reduction of some LOD schemes.*

NM-R8915 E.D. de Goede, J.H.M. ten Thije Boonkkamp. *NUMVEC FORTRAN library manual. Chapter: Partial differential equations. Routine: OEHPC.*

NM-R8916 E.D. de Goede. *Stabilization of a time integrator for the 3D shallow water equations by smoothing techniques.*

NM-R8917 J. Molenaar. *Non-linear multigrid in 2-D semiconductor device simulation: The zero current case.*

NM-R8918 B.P. Sommeijer, W. Couzy, P.J. van der Houwen. *A-stable parallel block methods.*

NM-R8919 A. Ivić, H.J.J. te Riele. *On the zeros of the error term for the mean square of* $|\zeta(\frac{1}{2}+it)|$.

NM-R8920 H.J.J. te Riele. *A new lower bound for the de Bruijn-Newman constant.*

NM-R8921 R.P. Brent, G.L. Cohen, H.J.J. te Riele. *Improved techniques for lower bounds for odd perfect numbers.*

NM-R8922 B. Koren, P.W. Hemker. *Damped, direction-dependent multigrid for hypersonic flow computations.*

NM-R8923 R.A. Trompert, J.G. Verwer. *A static-regridding method for two-dimensional parabolic partial differential equations.*

NM-R8924 W.H. Hundsdorfer, B.I. Steininger. *Convergence of linear multistep and one-leg methods for stiff nonlinear initial value problems.*

NM-N8901 R.M. Furzeland, J.G. Verwer. *CWI/Shell colloquium adaptive grid techniques for time-dependent partial differential equations.*

NM-N8902 7 J.G. Blom, J.G. Verwer. *On the use of the arclength and curvature monitor in a moving-grid method which is based on the method of lines.*

NM-N8903 M. Louter-Nool, D.T. Winter. *Benchmark of the initial release of the LAPACK Library.*

Publications in Journals, proceedings, etc.

NW1 J.G. Blom, J.M. Sanz-Serna, J.G. Verwer (1989). A Lagrangian moving-grid scheme for one-dimensional evolutionary partial differential equations. W.F. Ames, C. Brezinski (eds.). *IMACS Annals on Computing and Applied Mathematics,* in Section 3 of Vol. 1, J.C. Baltzer AG Scientific Publishing Company, Basel.

NW2 A. Granville, J. van de Lune, H.J.J. te Riele (1989). Checking the Goldbach conjecture on a vector computer. R.A. Mollin (ed.). *Number Theory and Applications,* Kluwer Academic Publishers, 423-433.

NW3 P.W. Hemker, B. Koren (1989). A non-linear multigrid method for the steady Euler equations. A. Dervieux, B. van Leer, J. Périaux, A. Rizzi (eds.). *Proceedings of the GAMM Workshop on the Numerical Simulation of Compressible Euler Flows* (Rocquencourt, 1986), Notes on Numerical Fluid Mechanics Vol. 26, Vieweg, Braunschweig, 175-196.

NW4 P.J. van der Houwen, B.P. Sommeijer (1989). The use of smoothing techniques in the method of lines. R. Vichnevetsky (ed.). *IMACS Annals on Computing and Applied Mathematics,* W.F. Ames, C. Brezinski (eds.). *Numerical and Applied Mathematics, Volume I,* J.C. Baltzer AG Scientific Publishing Company, Basel.

NW5 P.J. van der Houwen, B.P. Sommeijer (1989). Phase-lag analysis of implicit Runge-Kutta methods. *SIAM J. Numer. Anal. 26,* 214-229.

NW6 P.J. van der Houwen, B.P. Sommeijer (1989). Diagonally implicit Runge-Kutta-Nyström methods for oscillatory problems. *SIAM J. Numer. Anal. 26,* 414-429.

NW7 P.J. van der Houwen, B.P. Sommeijer, G. Pontrelli (1989). A comparative study of Chebyshev acceleration and residue smoothing in the solution of nonlinear elliptic difference equations. A. Bellen, C.W. Gear, E. Russo (eds.). *Numerical Methods for Ordinary Differential Equations,* Lecture

Notes in Mathematics 1386, Springer, Berlin, 69-96.

NW8 P.J. VAN DER HOUWEN, B.P. SOMMEIJER, P.A. VAN MOURIK (1989). Note on explicit parallel multistep Runge-Kutta methods. *J. Comput. Appl. Math. 27,* 411-420.

NW9 W.H. HUNDSDORFER, J.G. VERWER (1989). Stability and convergence of the Peaceman-Rachford ADI-method for initial-boundary value problems. *Math. Comp. 53,* 81-101.

NW10 W.H. HUNDSDORFER, J.G. VERWER (1989). Linear stability of the Hopscotch scheme. *Applied Numerical Mathematics 5,* 423-433.

NW11 W.H. HUNDSDORFER, J. SCHNEID (1989). On the equivalence of *BS*-stability and *B*-consistency. *BIT 29,* 505-511.

NW12 W.H. HUNDSDORFER (1989). Convergence of Runge-Kutta methods on classes of stiff differential equations. O. CELEBI, B. KARASÖZEN (eds.). *Proceedings of the International Symposium on Numerical Analysis,* September 1-4, 1987, Middle East Technical University, 89-98.

NW13 J. KOK (1989). Parallel programming with Ada. *The International Journal of Supercomputer Applications, Vol. 2, No. 4,* Winter 1988, 100-108.

NW14 J. KOK (1989). Parallel programming with Ada. J. DONGARRA, I. DUFF, P. GAFFNEY, S. MCKEE (eds.). *Vector and Parallel Computing, Issues in Applied Research and Development,* Ellis Horwood Ltd., Chichester, 189-203.

NW15 B. KOREN (1989). Euler flow solutions for transonic shock wave - boundary layer interaction. *Int. J. Numer. Methods in Fluids 9,* 59-73.

NW16 B. KOREN (1989). Upwind schemes for the Navier-Stokes equations. J. BALLMANN, R. JELTSCH (eds.). *Nonlinear Hyperbolic Equations - Theory, Numerical Methods and Applications* (Proceedings of the Second International Conference on Hyperbolic Problems, Aachen, 1988), Notes on Numerical Fluid Mechanics Vol. 24, Vieweg, Braunschweig, 300-309.

NW17 B. KOREN (1989). Upwind schemes, multigrid and defect correction for the steady Navier-Stokes equations. D.L. DWOYER, M.Y. HUSSAINI, R.G. VOIGT (eds.). *Proceedings of the 11-th International Conference on Numerical Methods in Fluid Dynamics* (Williamsburg, 1988), Lecture Notes in Physics Vol. 323, Springer, Berlin, 344-348.

NW 18 M. LOUTER-NOOL (1989). LINPACK routines based on Level 2 BLAS. *The Journal of Supercomputing 3,* 331-349.

NW19 C. NEBBELING, B. KOREN (1988/1989). Transonic shock wave - turbulent boundary layer interaction on a curved surface. *Delft Progr. Rep. 12,* 365-380.

NW20 H.J.J. TE RIELE (1989). Applications of supercomputers in mathematics. J.T. DEVREESE, P.E. VAN CAMP (eds.). *Scientific Computing on Supercomputers,* Plenum Press, 177-200.

NW21 H.J.J. TE RIELE, W.M. LIOEN, D.T. WINTER (1989). Factoring with the quadratic sieve on large vector computers. *J. Comp. Appl. Math. 27,* 267-278.

NW22 J.M. SANZ-SERNA, J.G. VERWER (1989). Stability and convergence at the PDE/stiff ODE interface. *Applied Numerical Mathematics 5,* 117-132.

NW23 J.M. SANZ-SERNA, J.G. VERWER (1989). Convergence analysis of one-step schemes in the method of lines. *Applied Mathematics and Computation, 31,* 183-196.

NW24 J.G. VERWER, J.G. BLOM, R.M. FURZELAND, P.A. ZEGELING (1989). A moving grid method for one-dimensional partial differential equations based on the method of lines. J.E. FLAHERTY (ed.). *SIAM Philadelphia,* 160-176.

NW25 J.G. VERWER, J.G. BLOM, J.M. SANZ-SERNA (1989). An adaptive moving grid method for one-dimensional systems of partial differential equations. *J. Comput. Phys. 82,* 454-486.

Other publications

NW26 B. KOREN (1989). *Multigrid and Defect Correction for the Steady Navier-Stokes Equations; Application to Aerodynamics.* Thesis, CWI, Amsterdam.

Department of Software Technology

Report series

CS-R8902 J.W. DE BAKKER. *Designing concurrency semantics.*

CS-R8903 F.W. VAANDRAGER. *A simple definition for parallel composition of prime event structures.*

CS-R8904 P. AMERICA, F.S. DE BOER. *Proving total correctness of recursive procedures.*

CS-R8905 A. PONSE. *Process expressions and Hoare's logic.*

CS-R8907 J.C.M. BAETEN, F.W. VAANDRAGER. *An algebra for process creation.*

CS-R8908 P.R.H. HENDRIKS. *Lists and associative functions in algebraic specifications - semantics and implementation - .*

CS-R8911 R.J. VAN GLABBEEK, W.P. WEIJLAND. *Branching time and abstraction in bisimulation semantics (extended abstract).*

CS-R8912 M. BEZEM. *Characterizing termination of logic programs with level mappings.*

CS-R8913 K.R. APT, R.N. BOL, J.W. KLOP. *On the safe termination of PROLOG programs.*

CS-R8917 R. BAGAI, M. BEZEM, M.H. VAN EMDEN. *On downward closure ordinals of logic programs.*

CS-R8918 H. WALINSKA DE HACKBEIL. *θ-government, thematic government and extraction asymmetries in zero derivation.*

CS-R8920 J.C.M. BAETEN, J.A. BERGSTRA. *Recursive process definitions with the state operator.*

CS-R8922 R.J. VAN GLABBEEK, W.P. WEIJLAND. *Refinement in branching time semantics.*

CS-R8923 Y. TOYAMA, J.W. KLOP, H.P. BARENDREGT. *Termination for direct sums of left-linear complete term rewriting systems.*

CS-R8924 H.J.M. GOEMAN. *Towards a theory of (self) applicative communicating processes: a short note.*

CS-R8926 J. HEERING, P.R.H. HENDRIKS, P. KLINT, J. REKERS. *The syntax definition formalism SDF - reference manual - .*

CS-R8927 L.C. VAN DER GAAG. *Belief networks in plausible reasoning.*

CS-R8928 K.R. APT, H.A. BLAIR. *Arithmetic classification of perfect models of stratified programs (extended version).*

CS-R8930 J.C.M. BAETEN, J.A. BERGSTRA, S. MAUW, G.J. VELTINK. *A process specification formalism based on static COLD.*

CS-R8931 P.J.F. LUCAS. *Multiple inheritance and exceptions in frame systems.*

CS-R8932 J.W. KLOP, A. MIDDELDORP. *Sequentiality in orthogonal term rewriting systems.*

CS-R8933 J.G. REKERS. *Modular parser generation.*

CS-R8934 J.C.M. BAETEN, J.A. BERGSTRA. *Design of a specification language by abstract syntax engineering.*

CS-R8940 H. WALINSKA. *The syntax of Slavic aspect (extended abstract).*

CS-R8942 R.N. BOL, K.R. APT, J.W. KLOP. *An analysis of loop checking mechanisms for logic programs.*

CS-R8943 D. TURI. *Logic programs with negation: classes, models, interpreters.*

CS-R8944 A. MIDDELDORP. *Confluence of the disjoint union of conditional term rewriting systems.*

CS-R8948 D. TURI. *Extending S-interpretations to logic programs with negation.*

CS-R8950 J.F. GROOTE. *Transition system specifications with negative premises.*

CS-R8951 F.S. DE BOER, J.N. KOK, C. PALAMIDESSI, J.J.M.M. RUTTEN. *Semantic models for a version of Parlog.*

CS-R8952 F.S. DE BOER, J.N. KOK, C. PALAMIDESSI, J.J.M.M. RUTTEN. *Control flow versus logic: a denotational and declarative model for guarded Horn clauses.*

CS-R8953 P.H.M. AMERICA, J.J.M.M. RUTTEN. *A parallel object-oriented language: design and semantic foundations.*

CS-R8954 J.W. DE BAKKER, J.J.M.M. RUTTEN. *Concurrency semantics based on metric domain equations.*

CS-R8955 J.J.M.M. RUTTEN. *Deriving models for bisimulation from structured operational semantics.*

CS-R8956 E.-R. OLDEROG, K.R. APT. *Using transformations to verify parallel programs.*

CS-R8957 A. PONSE, F.-J. DE VRIES. *Strong completeness for Hoare logics of recursive processes: an infinitary approach.*

CS-R8958 J.A. BERGSTRA, J. HEERING. *Which data types have ω-complete initial algebra specifications?*

CS-R8959 A. MIDDELDORP. *Termination of disjoint unions of conditional term rewriting systems.*

CS-N8901 R.J. VAN GLABBEEK, U. GOLTZ. *Partial order semantics for refinement of actions - neither necessary nor always sufficient but appropriate when used with care - .*

Publications in Journals, proceedings, etc.

AP1 P. AMERICA, J.W. DE BAKKER, J.N. KOK, J.J.M.M. RUTTEN (1989). Denotational semantics of a parallel object-oriented language. *Information and Computation, Vol. 83,* 152-205.

AP2 P. AMERICA, J.J.M.M. RUTTEN (1989). Solving reflexive domain equations in a category of complete metric spaces. *Journal of Computer and System Sciences, Vol 39,* no *3*, 343-375.

AP3 K.R. APT, R.N. BOL, J.W. KLOP (1989). On the safe termination of PROLOG programs. *Proceedings 6th International Conference on Logic Programming,* The MIT Press, 353-368.

AP4 J.C.M. BAETEN (1989). Algebra and communicating processes. *Proceedings AMAST Conference,* Iowa City, 35-38.

AP5 J.C.M. BAETEN, J.A. BERGSTRA, J.W. KLOP (1989). An operational semantics for process algebra. *Mathematical Problems in Computation Theory, Banach Center Publications Vol. 21,* 47-81, PWN-Polish Scientific Publishers, Warsaw.

AP6 J.C.M. BAETEN, J.A. BERGSTRA, J.W. KLOP, W.P. WEIJLAND (1989). Term rewriting systems with rule priorities. *Theoretical Computer Science 67,* 283-301.

AP7 J.C.M. BAETEN, R.J. VAN GLABBEEK (1989). Abstraction and empty process in process algebra. *Fundamenta Informaticae XII,* 221-242.

AP8 J.W. DE BAKKER (1989). Designing concurrency semantics. G.X. RITTER (ed.) *Information Processing 89,* Elsevier, 591-598.

AP9 J.W. DE BAKKER, P.C. TRELEAVEN (eds.) (1989). Special issue on PARLE: Conference on Parallel Architectures and Languages - Europe, 1987. *Parallel Computing 9,* 127-278.

AP10 J.A. BERGSTRA, J.W. KLOP (1989). Process theory based on bisimulation semantics. *Proceedings REX School/Workshop,* Noordwijkerhout, June 1988, LNCS 354, Springer, 50-122.

AP11 J.A. BERGSTRA, J.W. KLOP (1989). ACPt: a universal axiom system for process specification. M. WIRSING (ed.). *Algebraic Methods: Theory, Tools & Applications,* Workshop Passau 1987, LNCS 394, Springer, 447-464.

AP12 M. BEZEM (1989). Compact and majorizable functionals of finite type. *Journal of Symbolic Logic 54,* 271-280.

AP13 M. BEZEM (1989). Characterizing termination of logic programs with level mappings. *Proc. of the North American Conference on Logic Programming,* Cleveland, MIT Press, Cambridge, 69-80.

AP14 F.S. DE BOER (1989). Compositionality in the temporal logic of concurrent systems. E. ODIJK, M. REM, J.-C. SYRE (eds.). *Proc. PARLE 89,* LNCS 366, Springer, 406-423.

AP15 F.S. DE BOER, J.N. KOK, C. PALAMIDESSI, J.J.M.M. RUTTEN (1989). Control flow versus logic: a denotational and a declarative model for guarded Horn clauses. A. KRECZMAR, G. MIRKOWSKA (eds.). *Proceedings Mathematical Foundations of Computer Science, MFCS 89,* LNCS 379, Springer, 165-176.

AP16 F.S. DE BOER, J.N. KOK, C. PALAMIDESSI, J.J.M.M. RUTTEN (1989). Semantic models for a version of PARLOG. *Proceedings Int. Conference on Logic Programming,* MIT Press.

AP17 A. ELIËNS (1989). Extending Prolog to a parallel object-oriented language. M. COSNARD, C. GIRAULT (eds.). *Proc. Working Conference on Decentralized Systems,* IFIP WG 10.3,

North-Holland, 357-371.
AP18 L.C. VAN DER GAAG (1989). A conceptual model for inexact reasoning in rule-based systems. *Int. Journal of Approximate Reasoning, Vol. 3,* 239-258.
AP19 L.C. VAN DER GAAG (1989). Redeneren met onzekerheid I: Inleiding. *Informatie 31, no 11,* 814-819.
AP20 L.C. VAN DER GAAG (1989). Redeneren met onzekerheid I: Quasi-probabilistische modellen. *Informatie 31, no 12,* 981-990.
AP21 R.J. VAN GLABBEEK, U. GOLTZ (1989). Equivalence notions for concurrent systems and refinement of actions, Arbeitspapiere der GMD 366, Sankt Augustin, 1989. A. KRECZMAR, G. MIRKOWSKA (eds.). *Proceedings Mathematical Foundations of Computer Science, MFCS 89,* LNCS 379, Springer, 237-248.
AP22 R.J. VAN GLABBEEK, U. GOLTZ (1989). Partial order semantics for refinement of actions - neither necessary nor always sufficient but appropriate when used with care - . *Bulletin of the EATCS 38,* 154-163.
AP23 R.J. VAN GLABBEEK, F.W. VAANDRAGER (1989). Modular specifications in process algebra (with curious queues). M. WIRSING (ed.). *Algebraic Methods: Theory, Tools & Applications,* Workshop Passau 1987, LNCS 394, Springer, 465-506.
AP24 R.J. VAN GLABBEEK, W.P. WEIJLAND (1989). Branching time and abstraction in bisimulation semantics. G.X. RITTER (ed.). *Information Processing 89,* IFIP World Congress, San Francisco, Elsevier, 613-618.
AP25 R.J. VAN GLABBEEK, W.P. WEIJLAND (1989). Refinement in branching time semantics. *Proc. AMAST Conference,* Iowa City, USA, 197-201, also in *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.
AP26 J.F. GROOTE, F.W. VAANDRAGER (1989). Structured operational semantics and bisimulation as a congruence (extended abstract). G. AUSIELLO, M. DEZANI-CIANCAGLINI, S. RONCHI DELLA ROCCO (eds.). *Proc. ICALP 89,* Stresa, Italy, July 1989, LNCS 372, Springer.
AP27 J. HEERING, P.R.H. HENDRIKS, P. KLINT, J. REKERS (1989). The syntax definition formalism SDF-reference manual. *SIGPLAN Notices, Vol. 14 (11),* 43-75.
AP28 J. HEERING, P. KLINT, J. REKERS (1989). Incremental generation of parsers. *Proceedings of the SIGPLAN '89 Conference on Programming Language Design and Implementation,* 179-191.
AP29 J.W. KLOP, R.C. DE VRIJER (1989). Unique normal forms for the lambda calculus with surjective pairing. *Information and Computation, Vol. 80, no. 2,* 97-113.
AP30 P.J.F. LUCAS (1989). HEPAR: an expert system for the diagnosis of liver and biliary disease. *Liver 19 (5),* 261-275.
AP31 E.-R. OLDEROG (1989). Strong bisimilarity on nets: a new concept for comparing net semantics. *Proceedings REX School/Workshop,* Noordwijkerhout, June 1988, LNCS 354, Springer, 549-573.
AP32 J.J.M.M. RUTTEN (1989). Correctness and full abstraction of metric semantics for concurrency. *Proceedings REX School/Workshop,* Noordwijkerhout, June 1988, LNCS 354, Springer, 628-659.
AP33 W.P. WEIJLAND (1989). The algebra of synchronous processes. *Fundamenta Informaticae XII,* 139-162.

Other publications

AP34 P. AMERICA, J.J.M.M. RUTTEN (1989). A parallel object-oriented language: design and semantic foundations. J.W. DE BAKKER (ed.). *Languages for Parallel Architectures: Design, Semantics, Implementation Models,* Wiley 1-49.
AP35 K.R. APT, E.-R. OLDEROG (1989). A note on disjoint parallelism. *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.
AP36 J.C.M. BAETEN, F.W. VAANDRAGER (1989). An algebra for process creation. *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.
AP37 J.A. BERGSTRA, J.W. KLOP (1989). BMACP. *J.W. de*

Bakker, 25 jaar Semantiek, Liber Amicorum, CWI, Amsterdam.

AP38 J.A. BERGSTRA, J. HEERING, P. KLINT (eds.) (1989). *Algebraic Specification,* ACM Press, Addison-Wesley.

AP39 J.A. BERGSTRA, J.W. KLOP, A. MIDDELDORP (1989). *Termherschrijfsystemen,* Kluwer Programmatuurkunde.

AP40 J.W. DE BAKKER (ed.) (1989). *Languages for Parallel Architectures: Design, Semantics, Implementation Models,* Wiley.

AP41 J.W. DE BAKKER, W.-P. DE ROEVER, G. ROZENBERG (eds.) (1989). *Linear Time, Branching Time and Partial Order in Logics and Models for Concurrency,* Proceedings REX School/Workshop, Noordwijkerhout, June 1988, LNCS 354, Springer.

AP42 F.S. DE BOER (1989). A summary of the work on the proof theory for the language POOL. *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.

AP43 A. ELIËNS (1989). On the use of semantics: extending Prolog to a parallel object oriented language. *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.

AP44 R.J. VAN GLABBEEK, J.J.M.M. RUTTEN (1989). The processes of De Bakker and Zucker represent bisimulation equivalence classes. *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.

AP45 P.R.H. HENDRIKS (1989). Typechecking Mini-ML. J.A. BERGSTRA, J. HEERING, P. KLINT (eds.). *Algebraic Specification,* ACM Press, Addison-Wesley, 299-337.

AP46 P. KLINT (1989). Scanner generation for modular regular grammars. *J.W. de Bakker, 25 jaar Semantiek, Liber Amicorum,* CWI, Amsterdam.

AP47 J.N. KOK (1989). *Semantic Models for Parallel Computation in Data Flow, Logic- and Object-Oriented Programming.* Thesis, VU Amsterdam.

AP48 J. REKERS (1989). An implementation of SDF. J.A. BERGSTRA, J. HEERING, P. KLINT (eds.). *Algebraic Specification,* ACM Press, Addison-Wesley, 339-379.

AP49 J.J.M.M. RUTTEN (1989). *A Parallel Object-Oriented Language: Design and Semantic Foundations,* Thesis, VU Amsterdam, together with P.H.M. America, Philips Research.

AP50 J.W. SPEE (1989). *A Parallel Implementation of a Part of EQUIP in POOL2.* Prisma 0459.

AP51 J.W. SPEE (1989). *A Dynamic Load Balancing Strategy for POOL Programs.* Prisma 0464.

AP52 F.J. DE VRIES (1989). *Type Theoretical Topics in Topos Theory.* Thesis, RU Utrecht.

AP53 W.P. WEIJLAND (1989). *Synchrony and Asynchrony in Process Algebra.* Thesis, Universiteit van Amsterdam.

Department of Algorithmics and Architecture

Report series

CS-R8901 MING LI, P.M.B. VITÁNYI. *Kolmogorov complexity and its applications (revised version).*

CS-R8906 E. KRANAKIS, D.D.M. KRIZANC, L.G.L.T. MEERTENS. *Placing mirrors in grids.*

CS-R8914 L.G.L.T. MEERTENS. *Constructing a calculus of programs.*

CS-R8915 M. LI, P.M.B. VITÁNYI. *Inductive reasoning and Kolmogorov complexity (preliminary version).*

CS-R8916 M. LI, J. TROMP, P.M.B. VITÁNYI. *How to share concurrent wait-free variables.*

AM-R8918 M.J. COSTER. *Supercongruences.*

CS-R8919 M. LI, P.M.B. VITÁNYI. *A new approach to formal language theory by Kolmogorov complexity (preliminary version).*

CS-R8929 A.S. TANENBAUM, R. VAN RENESSE, H. VAN STAVEREN, G.J. SHARP, S.J. MULLENDER, A.J. JANSEN, G. VAN ROSSUM. *Experiences with the Amoeba distributed operating system.*

CS-R8935 E. KRANAKIS, D.D.M. KRIZANC. *Computing Boolean functions on anonymous networks.*

CS-R8936 L.M. KIROUSIS, E. KRANAKIS. *A survey of concurrent reader and writers.*

CS-R8939 J.T. TROMP. *How to construct an atomic variable.*

CS-R8945 J.N.E. Bos, M.J. Coster. *Addition chain heuristics.*
CS-R8949 E. Kranakis, M. Pocchiola. *Enumeration and visibility in integer lattices.*

Publications in Journals, proceedings, etc.

AA1 C.A. van den Berg, M.L. Kersten, P.M.G. Apers, P. America (1989). Language Support for Implementing a Main Memory Distributed Database System. *Proceedings of the Advanced Database Symposium,* Kyoto, Japan, 33-38.

AA2 I.J.P. Elshoff (1989). A distributed debugger for Amoeba. *Proc. Workshop on Parallel and Distributed Debugging,* SIGPLAN Notices 24, 1 (January 1989), 1-10.

AA3 M. Fokkinga (1989). Squiggolish derivation for partition problems—promotion as the driving force. *Lecture Notes International Summerschool on Constructive Algorithmics.*

AA4 P. Grefen, A. Wilschut, J. Flokstra, P.M.G. Apers, M.L. Kersten (1989). Implementing PRISMA/DB in an OOPL. *Proceedings 6th Int. Workshop on Database Machines,* Springer-Verlag, June 1989.

AA5 J.T. Jeuring (1989). The derivation of an algorithm for finding palindromes. *Lecture Notes International Summerschool on Constructive Algorithmics.*

AA6 J.T. Jeuring (1989). Deriving algorithms on binary labelled trees. P.M.G. Apers, D. Bosman, J. van Leeuwen (eds.). *Proceedings Computing Science in the Netherlands 1989.*

AA7 M.L. Kersten (1989). Using logarithmic code-expansion to speedup index access and maintenance. *Proceedings 3rd Int. Conf. Foundations of Data Organziation and Algorithms,* Paris, June 1989, 228-232.

AA8 E. Kranakis, L.G.L.T. Meertens (1989). Optimal packing of similar triangles. *EATCS Bull. 38,* 181-185.

AA9 L. Meertens (1989). Constructing a calculus of programs. *Proceedings Mathematics of Program Construction Conference,* LNCS 375, Springer-Verlag, 66-90.

AA10 L. Meertens (1989). The generic theory of binary structures; variations on trees; indeterminacy; observations on the 'squiggol spirit'. *Lecture Notes International Summerschool on Constructive Algorithmics.*

AA11 Ming Li, P.M.B Vitányi (1989). How to share concurrent asynchronous wait-free variables. *Poceedings ICALP 89,* LNCS 372, Springer-Verlag, 488-505.

AA12 Ming Li, P.M.B Vitányi (1989). Inductive reasoning and Kolmogorov complexity. *Poceedings 4th IEEE Structure in Complexity Theory Conference,* 165-185.

AA13 Ming Li, P.M.B Vitányi (1989). A new approach to formal language theory by Kolmogorov complexity. *Poceedings ICALP 89,* LNCS 372, Springer-Verlag, 506-520.

AA14 Ming Li, P.M.B Vitányi (1989). A theory of learning simple concepts under simple distributions and avarage case complexity for the universal distribution. *Poceedings 30th IEEE Symposium on Foundations of Computer Science,* 34-39.

AA15 Ming Li, P.M.B Vitányi (1989). A theory of learning simple concepts under simple distributions, *Poceedings 2nd ACM Workshop on Computational Learning Theory,* 386.

AA16 J. Tromp (1989). How to construct an atomic variable. J.C. Bermond, M. Raynal (eds.). *Proceedings 3rd International Workshop On Distributed Algorithms,* Nice, France, September 1989, LNCS 392, 292-302.

Other publications

AA17 C.A. van den Berg (1989). *Translating SQL Update Statements into XRA.* PRISMA Report P448.

AA18 C.A. van den Berg (1989). *PRISMA/DB1 Recovery.* PRISMA Report P518.

AA19 L. Meertens (1989). *Epis User Manual* (beta release), CWI.

AA20 Miguel Carrasquer, Denise Draper, Lambert Meertens, Steven Pemberton (1989). *Euromath Deliverable: Func-*

tional Specification.
AA21 MING LI, P.M.B VITÁNYI (1989). *A Theory of Learning Simple Concepts under Simple Distributions and Avarage Case Complexity for the Universal Distribution.* ITLI Techn. Rept., Computation and Complexity Theory Series CT-89-07, University of Amsterdam, October.
AA22 MING LI, P.M.B VITÁNYI (1989). *Applications of Kolmogorov Complexity in the Theory of Computation.* Techn. Report CS-89-01, Computer Science Department, York University, Ontario, Canada, March.
AA23 MING LI, P.M.B VITÁNYI (1989). *Inductive Reasoning and Kolmogorov Complexity.* Techn. Report CS-89-05, Computer Science Department, York University, Ontario, Canada, May.
AA24 S.J. MULLENDER (1989). *Distributed Systems.* ACM Press, ISBN 0-201-41660-3.
AA25 S.J. MULLENDER (1989). *An Advanced Course on Distributed Computing.* Fingerlakes'89 Lecture Notes, Cornell University.
AA26 N.TH. VERBRUGGE (1989). *Translation of Nested SQL Queries into Extended Relational Algebra.* PRISMA Report P501.
AA27 N.TH. VERBRUGGE (1989). *Practical Use of Tuple Identifiers in PRISMA/DB.* PRISMA Report P502.
AA28 L. VAN DER VOORT (1989). *Working Paper on Tropics SQL.* TROPICS Report.

Department of Interactive Systems

Report series

CS-R8909 A.A.M. KUIJK, E.H. BLAKE. *Faster Phong shading via angular interpolation.*
CS-R8921 I. HERMAN. *2.5 Dimensional graphics systems.*
CS-R8938 I. HERMAN. *On the projective invariant representation of conics in computer graphics.*
CS-R8946 D.A. DUCE, R. VAN LIERE, P.J.W. TEN HAGEN. *An approach to hierarchical input devices.*
CS-R8947 D.A. DUCE, R. VAN LIERE, P.J.W. TEN HAGEN. *Components, frameworks and GKS input.*
CS-R8960 A.A.M. KUIJK. *Temporal issues of animate response.*

Publications in Journals, proceedings, etc.

IS1 V. AKMAN, P.J.W. TEN HAGEN (1989). The power of physical representations. *AI Magazine, Vol. 10, No. 3,* 49-65.
IS2 E.H. BLAKE, P. WISSKIRCHEN (1989). Object-oriented graphics. W. PURGATHOFER, J. SCHÖNHUT (eds.). *Advances in Computer Graphics V,* Springer-Verlag, Berlin, 109-154.
IS3 D.A. DUCE, R. VAN LIERE, P.J.W. TEN HAGEN (1989). Components, frameworks and GKS input. W. HANSMANN, F.R.A. HOPGOOD, W. STRASSER (eds.). *Proceedings Eurographics '89 Conference,* North-Holland, Amsterdam, 87-106.
IS4 I. HERMAN (1989). 2.5D Graphics systems. W. HANSMANN, F.R.A. HOPGOOD, W. STRASSER (eds.). *Proceedings Eurographics '89 Conference,* North-Holland, Amsterdam, 415-426.
IS5 I. HERMAN (1989). On the projective invariant representation of conics in computer graphics. *Computer Graphics Forum, Vol. 8, No. 4,* 301-314.
IS6 A.A.M. KUIJK, E.H. BLAKE (1989). Faster Phong shading via angular interpolation. *Computer Graphics Forum, Vol. 8, No. 4,* 315-324.
IS7 A.A.M. KUIJK, E.H. BLAKE (1989). Faster Phong shading via angular interpolation. *Proceedings of International Conference on Computer Aided Design & Computer Graphics,* August 10-12, 1989 Beijing, China, International Academic Publishers, 35-41.
IS8 D.B.M. OTTEN, P.J.W. TEN HAGEN (1989). On the role of delegation and inheritance in database management systems. *Proceedings of the Third IFIP WG 5.2 Workshop on Intelligent*

CAD, International Federation for Information Processing, Working Group 5.2, Japan, 340-360.

IS9 J.L.H. ROGIER, F. TOLMAN (1989). De formulering van een ontwerpparadigma voor bouwkundig ontwerpers. *Proceedings CAPE '89,* Samsom, 383-395.

IS10 J.L.H. ROGIER, P.J. VEERKAMP, P.J.W. TEN HAGEN (1989). An environment for knowledge representation for design. *Proceedings of Civil Engineering Expert Systems,* Escuela Tecnica Superior de Ingenieros de Caminos, Canales y Puertos.

IS11 JANTIEN VAN DER VEGT, HANS BUFFART, CEES VAN LEEUWEN (1989). The Structural Memory: A network model for human perception of serial objects. *Psychological Research 50,* Springer-Verlag, Berlin, 211-222.

Other publications

IS12 M. BAKKER (1989). *GKS/C, The Graphical Kernel System, Language Bindings, Part 4: C,* ISO/IEC Draft Proposal 8651-4.

IS13 M. BAKKER (1989). *GKS-3D/C, The Graphical Kernel System for Three Dimensions Language Bindings, Part 4: C,* ISO/IEC Draft Proposal 8806-4; ISO/IEC Interim Draft International Standard 8806-4.

IS14 E.H. BLAKE (1989). Review of 'Image Synthesis—Elementary Algorithms' by G. Hégron. *Computer Graphics Forum, Vol. 8,* 359.

IS15 I. HERMAN (1989). Book Review on: 'Computer Graphics Hardware, Image Generation and Display - A Tutorial, IEEE Publications'. *Computer Aided Design,* 660.

IS16 I. HERMAN, P.J.W. TEN HAGEN, M.M. DE RUITER (1989). *Interface Specification in DFC,* Dataflow project internal specification document, 50 pages.

IS17 J.L.H. ROGIER (1989). *The Use of STEP in an Intelligent Design System for Architectural Design,* Preliminary Proceedings of the Third Eurographics Workshop on Intelligent CAD Systems, 45-56.

IS18 P.J. VEERKAMP, R.S.S. PIETERS KWIERS, P.J.W. TEN HAGEN (1989). *Design Process Representation in ADDL,* Preliminary Proceedings of the Third Eurographics Workshop on Intelligent CAD Systems, 115-130.