

CST Bulletin 7

Mededelingen van Computer Systemen en Telematica
voor de computergebruikers aan het CWI.
Verschijnt onregelmatig.
Dit is aflevering 7 dd. 2 maart 1992.



INHOUD:

1. First Eastern Europe, then Russia, now CST	Dick Buterman
2. Enige recente wijzigingen in het machinepark	Sjoerd Mullender
3. Nieuwe groep binnen CST	Peter Dijkhuis
4. Window systems at CWI	Robert van Liere
5. Opnieuw een <i>boring</i>	Frank Kuiper
6. Password Security Revisited	Daniel Karrenberg
7. Van oude en nieuwe CST Bulletins	A.C. IJsselstein
8. Enkele administratieve meldingen	Frank Kuiper
9. De nieuwe situatie bij de SGI's	Sjoerd Mullender

1. First Eastern Europe, then Russia, now CST: changes in the last twelve months.

The masthead of the CST Bulletin contains, among other things, the phrase: *Verschijnt onregelmatig*. If we study the publication schedule of the CST Bulletin, it is clear that our promise to provide information at non-regular intervals has been fulfilled. Unfortunately, the interval between issues 6 and 7 of the Bulletin has been longer than even we had hoped: our goal of one issue per half year did not take into account a number of fundamental changes that have occurred within CST (and within CWI!) during the past 12 months. Starting with this issue, we hope to begin a semi-annual production schedule that will provide background information to the CWI user community on issues related to the support of the hardware and software computing infrastructure at CWI.

This issue of the CST Bulletin reviews some of the major changes in people, hardware and software that have taken place within CST during the past year. All of these changes were made to increase the effectiveness of the computational support environment at CWI. In this introduction, I will outline some of the major policy-related aspects of these changes, and point readers to the relevant other articles in this issue that provide additional implementation details.

First, one result of the recent CWI reorganization has been the expansion of CST. In addition to our existing hardware and systems software support and research tasks, we now also have the responsibility for providing general software and user services support to all users at CWI. Internally, CST is now organized as three groups: a research group (that studies systems aspects of multi-media information manipulation), a systems

and networks support group (that provides hardware and operating systems supports for our central servers, workstations and local, national and international network infrastructure), and a user services group (that provides support for all classes of user problems associated with using our workstation-based facilities). As part of this reorganization, Frank Kuiper and Peter Dijkhuis have accepted positions as coordinators of the systems and networks, and user services groups, respectively. Although much of this issue contains information relating to CST's traditional systems and networks support activities, Peter Dijkhuis has also contributed a short description of our plans for providing user services support: *Nieuwe groep binnen CST*. (More information will be made available in the coming months on organizational issues within CST.) Our final change on the personnel front was the addition of Lynda Hardman to our research staff. Lynda joined CST in January as a researcher in a PBTS-subsidized project on multimedia-based management games. Lynda, whose area of expertise is hyperdata modelling, comes to CWI from OWL, a company best known for its *Guide* hypertext product.

In addition to reorganizing its people, CST also spent considerable effort in reorganizing CWI's central computing facilities during the past year. The first goal of these changes was to increase the effectiveness of our (shrinking) systems support staff by reducing the number of different types of architectures that needed to be supported. As a result, we now support two major lines of central computers (Sun Microsystems SPARC-based servers and Silicon Graphics MIPS-based PowerCenters) instead of the six that had been supported. (A trivia question for CST Bulletin readers: can you name the architecture types and year-of-introduction of these six?) Our second goal was to increase the efficiency of our central compute servers by providing a more common software base that would attract a larger per-machine user community. As a consequence, we replaced our existing Alliant FX-4 and Encore Multimax-520 with a SUN OS-compatible FPS 510 "supercomputer." The FPS machine, named *boring.cwi.nl* has a fast SPARC CPU and 28 i860 processors arranged in a 4x7 matrix. It is described by Frank Kuiper in an article

starting on page 7. The other changes in our central computing environment are described by Sjoerd Mullender in *Enige recente wijzigingen in het machinepark*.

Although some dramatic changes were made to our central facilities this year, these were not the only changes that were made to our general infrastructure. This past summer, CWI became one of the first customers of a new Silicon Graphics workstation: the Indigo. The Indigo is a (relatively!) low-cost workstation that combines impressive graphics and audio capabilities with first-rate processor performance. The Indigo was selected after our annual review of the offerings from Sun Microsystems, HP/Apollo and Digital. We felt that the Indigo represented the first of a new generation of workstations, in which the facilities provided across the user/computer interface received the same attention as has traditionally been provided to the system/processor interface in other workstations. Along with the purchase of 20 Indigos for expanding our research support, we also upgraded and traded-in 14 of our older Silicon Graphics workstations, making SGI (as Silicon Graphics is known in the trade) our second most prevalent supplier of workstations and servers, behind Sun Microsystems. Sjoerd Mullender, who has primary responsibility for supporting our SGI equipment, provides an overview article on these changes in *De nieuwe situatie bij de SGI's*.

Finally, this issue provides several articles of general interest that are not directly related to changes in our staff or equipment facilities. Robert van Liere (of IS) provides a global review of the window systems available at CWI. Daniel Karrenberg, who will shortly be leaving CWI for a one-year appointment as coordinator of the RIPE networking organization, concludes a series of articles on password security. Lastly, A.C. IJsselstein—the patient editor of the CST Bulletin—provides information on how users can gain access to former issues of this irregular (but hopefully informative) publication.

Dick Bulterman (*dcab@cwi.nl*)



2. Enige recente wijzigingen in het machinepark

De laatste tijd hebben er nogal wat veranderingen plaats gevonden in het machinepark van het CWI. Diverse machines zijn verdwenen en er zijn ook allerlei nieuwe machines binnen gekomen.

VERDWENEN MACHINES

Piring was een Harris HCX-9. Toen *piring* kwam, was het een Harris HCX-7, maar deze machine is een paar jaar geleden omgebouwd tot de snellere HCX-9. In de beginstijd heeft *piring* nogal veel problemen gegeven, maar de laatste jaren was de machine behoorlijk stabiel. Lange tijd was het de snelste machine op het CWI, maar de workstation technologie is nu zover gevorderd dat dit niet meer geldt.

Paring was een Alliant FX-4. *Paring* is in september 1988 gekomen. De machine had 5 processoren, een I/O processor en 4 Computational Elements. De machine was vooral bedoeld voor de afdeling Numerieke Wiskunde. De vervangende machine is het nieuwe FPS systeem dat we *boring* genoemd hebben (meer hierover in het artikel *Opnieuw een boring* van Frank Kuiper elders in dit CST Bulletin).

Curing was een Encore Multimax 520. *Curing* kwam, evenals *paring*, in september 1988. Het was een machine met 8 CPU's en daarmee de snelste machine voor compilaties. Het nadeel was natuurlijk dat die gecompileerde software nergens anders kon draaien. *Curing* deed dienst als *compute server*. Dat wil zeggen, mensen logden in op *curing* om daar te werken. *Curing* is vervangen door *poseidon*.

Turing was een Encore Multimax 320. *Turing* was als tweedehands machine overgenomen van Brown University in Providence (Rhode Island). *Turing* heeft vooral dienst gedaan als *file server* voor de Silicon Graphics workstations.

ENKELE NIEUWE MACHINES

Poseidon is een Silicon Graphics 4D260S. Het is een machine met 6 MIPS R3000 processoren die op 25 MHz draaien. Er zit 64

megabyte geheugen in en er is ongeveer tweeëneenhalf gigabyte disk-ruimte. *Poseidon* is bedoeld als *compute server* met dezelfde taak als *curing* had.

De Silicon Graphics Iris Indigo's zijn het nieuwe type workstation van Silicon Graphics. Een andere benaming voor deze machines is 4DRPC. Het voorvoegsel *4D* geldt voor alle machines van Silicon Graphics. Het gedeelte *RPC* staat voor Risc PC. De Indigo's hebben een MIPS R3000 CPU die met een kloksnelheid van 33 MHz draait. Dat is bijna even snel als de 4D35's die we al hadden (36 MHz). De Indigo's hebben 8 bit planes, dat wil zeggen, er kunnen maximaal 2^8 verschillende kleuren op het scherm vertoond worden. De software merkt daar echter niet veel van. In tegenstelling tot de oudere machines van Silicon Graphics worden de grafische mogelijkheden voor het grootste deel in software gedaan. Oudere systemen deden een belangrijk deel in hardware. Er zijn 35 Indigo's gekocht, die nu vrijwel allemaal geïnstalleerd zijn. De Indigo's heten naar bloemen.

Sjoerd Mullender (sjoerd@cwi.nl)



3. Nieuwe groep binnen CST.

Met ingang van 1 januari 1992 is STO opgeheven en daarmee ook de groep Computer Service en Ondersteuning (CSO). De gebruikersvoorlichting en -ondersteuning wordt vanaf die datum verzorgd door de nieuw ingestelde Users Services Group (USG) binnen CST. Voorlopig behoren A.C. IJsselstein, Frank van de Wiel, Kees van't Hoff, Michiel de Vries en Peter Dijkhuis (coördinator) tot die groep. Dit betekent dat de gehele voorlichting (hardware en software) onder auspiciën van CST uitgevoerd zal worden. Parallel aan deze organisatorische wijzigingen loopt de nieuwe, meer gestructureerde opzet van de gebruikersvoorlichting. Op dit moment wordt daar hard aan gewerkt.

Onderdelen van de nieuwe opzet zullen zijn:

- vaste informatiekanalen naar gebruikers

(CST Bulletin, News);

- vast herkenbaar eerste-lijnsvoorlichtingspunt: bemand op vaste tijden, vast telefoonnummer en pieper. Aanmeldingspunt voor verzoeken om tweede-lijns voorlichting etc.
- inventarisatie software binnen CWI en indeling van nieuw beschikbaar gestelde software in ondersteuningscategorien;
- aanmeldingsprocedure nieuwe software;
- centrale organisatie manuals: zowel manuals 'op de kamers' als een of twee vaste 'inkijkplaatsen';
- actievere begeleiding nieuwkomers;
- verzorging van op de situatie toegesneden handleidingen en cursussen.

Het kan zijn dat u in deze periode benaderd bent of zal worden met vragen over uw gebruik (en wensen) van hard- en software binnen het CWI.

We houden u op de hoogte via de reeds eerder genoemde informatiekanalen.

Peter Dijkhuis (peter@cwi.nl)



4. Window systems at CWI

This article give a short overview of the X window system, the de facto standard window system used at the CWI. It provides some technical background on how the X window system works and refers to local X gurus where additional information can be obtained.

We are assuming that you have already had some initial experience in using X. If, for some reason you can't get the window system running, just ask Frank van der Wiel (frankw@cwi.nl) for help in installing your initial environment.

HOW DOES THE X WINDOW SYSTEM WORK?

The X window system, or X for short, is a network-transport window system. With X, multiple applications can run simultaneously in windows, generating text and graphics on a bitmap display. Network transparency means that application programs can run on other machines scattered throughout the network.

X provides facilities for generating multifont text and two dimensional graphics in a hierarchy of rectangular windows. Every window can be thought of as a *virtual screen* and can contain subwindows within it. Windows can overlap each other like stacks of papers on a desk and can be moved, resized and re-stacked dynamically. Applications commonly use several hundred windows; for example, windows are used to implement individual user interface components such as scroll bars, menus, buttons, etc.

Although end users typically think of themselves as a client of the system, in network terms, X applications are clients which use the network services of the window system. A program running on the workstation with the display hardware provides these services and so is called the X server. The X server acts as an intermediary between applications and the workstation, handling output from the clients to the display and forwarding input (entered with a keyboard or mouse) to appropriate clients for processing.

Clients and servers use some form of inter-process communication to exchange information. The syntax and semantics of this conversation are defined by a communication protocol. This protocol is the foundation of the X window system. Clients use the protocol to send requests to the server to create and manipulate windows, to generate text and graphics, to control input from the user, and to communicate with other clients. The server uses the protocol to send information back to the client in response to various requests and to forward keyboard and other user input to the appropriate clients.

The X protocol is designed to be extensible without requiring incompatible changes to

existing applications. This means that the base protocol can be extended with additional functionality without having any influence on already existing X clients. Examples of protocol extensions are additional graphics models (i.e. PEX and DPS) and real time video (VEX).

Complicated? Don't worry, nobody at the CWI really understands the X protocol in depth! Robert van Liere (robertl@cwi.nl) has spent some time in writing a X server for some exotic graphics device. If you want some details about the internals of the X protocol, ask him.

LIBRARIES AND TOOLKITS.

Although X is fundamentally defined by a network protocol, most application programmers do not want to think about bits, bytes and message formats. Therefore, X has an interface library which provides a familiar procedural interface that masks the details of the protocol encoding and transport interactions, and automatically handles the buffering of requests for efficient transport to the server. The exact interface for this library differs for each programming language; *Xlib* is the library for the C programming language.

In X, many facilities that are built into the other window systems are provided by client libraries. The X protocol does not specify menus, scroll bars, and dialog boxes or how an application should respond to user input. The protocol and X library avoid mandating such policy decisions as much as possible and should be viewed as a construction kit providing a rich set of mechanisms that can implement a variety of user interface policies. It is important to keep in mind that the X library is only one layer in a complete X programming environment. As such, the X protocol is correctly viewed as just one component in an overall distributed systems architecture, not as the complete architecture itself.

Higher level layers, called toolkits, have been designed and implemented which hide various forms of tedium from the programmer. Two major user interface toolkits are *Motif* and *OpenLook*. Applications using ei-

ther can coexist simultaneously. Both of these can use the same X toolkit structure, though they provide quite different results to the end user. Significant research toolkits include *InterViews*, written in C++ at Stanford, the *Andrew Toolkit* of CMU and several Common Lisp toolkits.

At the CWI we have, among others, the Motif, OpenLook, *Xaw*, InterViews toolkits. No one person knows the details of all these toolkits, although you can try your luck with Guido van Rossum (guido@cwi.nl) or Robert van Liere (robertl@cwi.nl). In fact, Guido has designed and implemented his own toolkit, called STDWIN, which has been ported to a vast number of platforms.

On still a higher level there are a number of user interface management systems and other application builders for X. All of this, of course, is to enable applications to be built easily and cheaply. Application builders are now appearing in quantity for X. They allow a user to define the layout of the user interface and then generate the corresponding code that is needed by the application program.

A SPECIAL CLIENT: THE WINDOW MANAGER.

A user interface can be viewed as having two primary components: the interaction with the user that is logically internal to an application (for example, typing text into a text editor or changing a cell's content in a spreadsheet), and the interaction that is logically external to an application (for example, moving and resizing an application window or turning an application window into an icon). The external user interface is hardwired into many other window systems, but this is not the case with X. The X protocol does not define an external user interface at all. Rather, the protocol provides mechanisms with which a variety of external user interfaces can be built. These mechanisms are designed so that a single client, called the window manager, can provide the external user interface independent of all of the other clients. A window manager can automatically:

- provide title bars, borders, and other window decorations for each application,

- provide a uniform means of moving and resizing windows,
- enforce a strict window layout policy if it desires,
- provide uniform icons for applications,
- provide a uniform interface for switching the keyboard between applications.

With a suitable set of conventions, which have been standardized and called the *Inter-client Communication Convention*, applications are insensitive to the external user interface provided by a window manager and run correctly unmodified in multiple environments.

At the CWI we have three very popular window managers: *twm*, *mwm* and *4Dwm*. Which you opt for is really a matter of taste. All three need *startup* files, called *rc*-files, which are located in your home directory. The easiest is probably to use the default window manager *twm*. If, for some reason you have difficulties with your *rc*-files, try Frank van der Wiel (*frankw@cwi.nl*) for help.

RESOURCES.

Within the X library there is also a resource manager which provides a set of tools for specifying and manipulating user preferences (e.g. geometry, colors, fonts). Resources, also referred to as *defaults*, are frequently used to control the appearance or function of a particular program or subsystem. They also provide a convenient way to tailor whole collections of applications with a minimal amount of work.

In X, this is done by using the window property mechanism in the X protocol to store resources in the server, where they are available to all clients. As a result, defaults may now be dynamically specified based on the particular workstation being used. This is particularly useful in setting up different defaults for monochrome and color workstations or for setting up defaults for displays of different sizes.

Although this is all very powerful, figuring out that this can be specified at all, let alone

how, is currently a problem with the documentation. Eventually, all user interface components should be documented just as UNIX commands are today. Application documentation would then only need to describe how user interface components are combined. Until then, the best places to look for information on which resources may be specified are:

- i) the manual pages for the application,
- ii) any documentation for the widgets used by the application,
- iii) your neighbours resource file!

The resource manager related facilities are quite complicated. See Frank van der Wiel (*frankw@cwi.nl*) for details if you have any problems in specifying your resource files.

A LAST PIECE OF ADVICE.

X is a tremendously large and complex system (in fact, some people refer to it as a black hole...). It is extremely flexible and can be tailored in very many ways. **WARNING: IT TAKES A NOVICE YEARS TO LEARN TO PROGRAM IN X AT AN ACCEPTABLE LEVEL.** Don't try it, unless you have to. There are very knowledgeable X users at the CWI. Try to formulate what you want from the system and go to one of these gurus and ask for their advice. This will probably save you lots of time. If you can't find anybody that can help, try to post your question on newsgroups like *cwi.sun* or *cwi.sgi*.

Robert van Liere, Departement of Interactive Systems, (*robertl@cwi.nl*)



5. Opnieuw een boring.

Zoals velen misschien al gemerkt hebben, heeft het CWI sinds enige tijd de beschikking over een zogenaamde 'Super Mini' computer. Dit is de benaming voor een machine waar-

van de rekenkracht ver boven die van een mini-computer uitkomt en in dezelfde orde van grootte ligt als die van een super-computer, maar die de uiterlijke kenmerken en karakteristieken heeft van een mini-computer.

Boring is zo'n machine. Om preciezer te zijn, *boring* is een FPS 510A2. Deze FPS (Floating Point Systems) bestaat in feite uit twee geïntegreerde architecturen binnen een machine. Ten eerste is daar de reeds vertrouwde omgeving van een Sparc processor, welke compatibel is met de Sun Sparc's die het CWI heeft. Het *operating system* op deze processor is compatibel met SunOS (met SunOS 4.1.1 om precies te zijn). Wij moeten nog een tweede Sparc processor ontvangen, maar de levering hiervan is enkele maanden vertraagd, i.v.m. de overname van FPS door Cray Computing. (Kan het CWI straks zeggen dat ze een 'Cray' in huis heeft :-).)

De tweede architectuur is veel interessanter en de reden dat deze machine aangeschaft is. Deze tweede architectuur bestaat uit in totaal 28 Intel i860 processoren, gerangschikt in een 7x4 matrix (fig. 1). Elk processor haalt

80 MegaFlops bij single-precision operaties! Verbinding met de rest van het systeem wordt verzorgd door een 64-bit brede bus, de zogenaamde SIA (*Scalable Interconnect Architecture*), waarbij een meervoudige, gelijktijdige benadering van het geheugen mogelijk is. Binnen de multiprocessor architecturen zijn er twee uitersten te onderscheiden: A) alle processoren op een bus (enigzins vergelijkbaar met machines aan een ethernet); B) alle processoren zijn verbonden in een uitgebreid ster-netwerk, waarbij elke processor direct met elke andere processor verbonden is. Deze laatste methode levert zeer snelle machines op, maar is ook uiterst kostbaar. De eerste methode is, hoewel relatief goedkoop, beperkt in mogelijkheden en snelheid. De situatie zoals we die dus nu hebben in *boring* is daarbij een compromis. Vermeld dient nog te worden dat het aantal processoren per bus-aansluiting uitgebreid kan worden tot twaalf, zodat maximaal 84 processoren geplaatst kunnen worden.

Aan al dit moois kleeft nog enigzins een bazaar. Het gebruik van de 28 Intel i860 processoren is alleen mogelijk met behulp

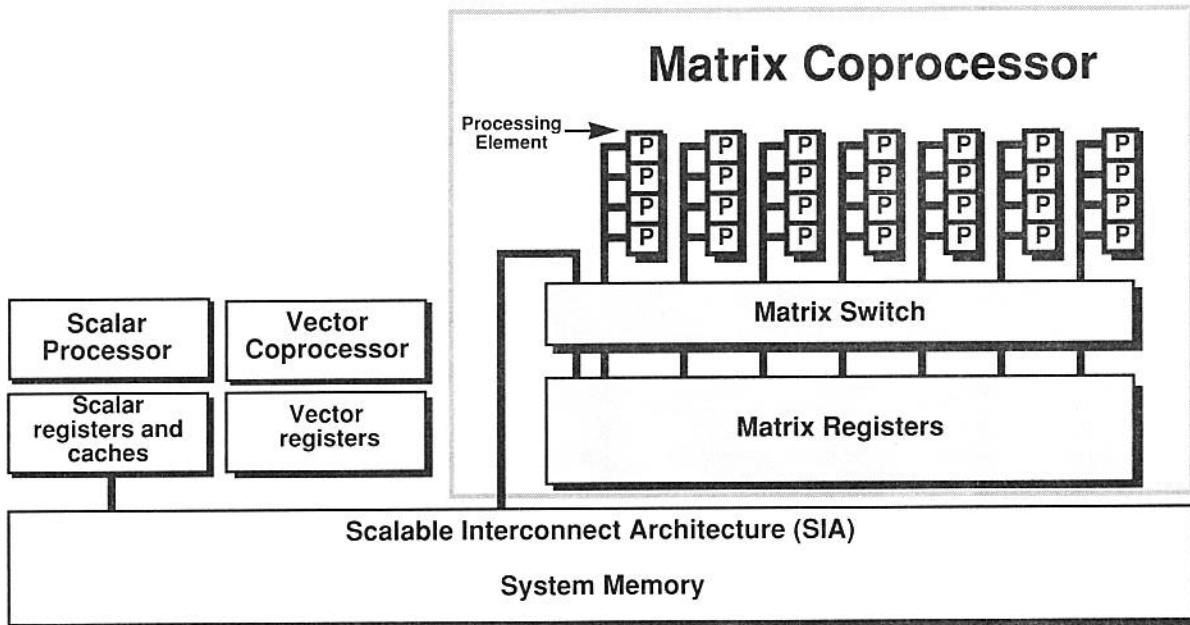


Fig. 1: Matrix Coprocessor Architecture

van een aparte Fortran-compiler en de programma's moeten via een batch-job systeem aan de multiprocessor pool worden doorgegeven. In een nieuwe versie van de het operating system kan ook in *multi user mode* op de multiprocessoren gewerkt worden. Een speciale compiler voor C is op wat langer termijn in de planning.

Diegenen die eens een keer wat willen uitproberen op deze machine kunnen zich bij ondergetekende vervoegen voor een account.

Frank Kuiper (frankk@cwi.nl)



6. Password Security Revisited.

In *CST Bulletin 4* (March 1990) we tried to get make CWI users aware of the issues involved in password security. Attempts to gain unauthorised access to Internet systems are on the increase. Thus we would like to stress again that passwords like

processi
nathalie
whiskey
gandalf
rosebud
Telefoon
dharma

are no obstacle to any but the most simple-minded intruder. These examples were taken from those passwords found by a simple program run from time to time over the last few weeks.

Please do not choose passwords which can be found in any dictionary. Use the passphrase method described in the earlier article in *CST Bulletin 4*. The relevant part of that article is displayed on this page at the top of the next column.

Daniel Karrenberg (dfk@cwi.nl)



All words you can find in a dictionary are not particularly safe as passwords. Also short passwords chosen from a limited character set are prone to be found by trial and error. So here are a few rules on what passwords should look like:

- longer than 6 characters
- upper case and lower case and numeric
- not a name not a word found in a dictionary
- not the username itself
- not well known acronyms (CWI, MC, NWO, O&W)

So *NilvD2rsg* would be a good choice. Now it looks very difficult to remember such gibberish. My personal way of overcoming this is to choose a passphrase rather than a pass-word and make up the password from that phrase:

Now it looks very Difficult 2 remember such gibberish is a good example. There are endless variations on this theme. For instance, you need not take the first letter from each word.

The description of the *passphrase method* in *CST Bulletin 4* (see the Daniel Karrenberg's article *Password Security Revisited* in this issue on this same page).

7. Van oude en nieuwe CST Bulletins.

Dit Bulletin draagt het nummer 7, en er zijn dus zes voorafgaande nummers. Wat daarin staat is veelal verouderd, maar er zijn toch ook nog wel wat artikelen met nu nog relevante informatie (bijv. het artikel *De printer banner page* van de hand van Sjoerd Mulender in *CST Bulletin 4* van 8 maart 1990). In de onderstaande alinea's volgt enige informatie over de bereikbaarheid van die informatie.

In elk van de terminal-ruimtes M281 en M377 bevindt zich een dunne bruine CWI-

ringband (2-gats, met rug-tekst *CST Bulletin*) waarin alle verschenen nummers te vinden zijn. Bovendien heeft de Bibliotheek er voor gezorgd dat het Bulletin op de informatica-planken van haar tijdschriftenzaal te vinden is. We hadden inderdaad ook een klein aantal losse nummers per oplage achter de hand gehouden om alsnog eventuele geïnteresseerde *collectioneurs* te bedienen. Maar helaas bleek kort voor het in functie treden van onze nieuwe secretaresse, dat dit bestand een niet te negeren overeenkomst was gaan vertonen met het bij wiskundigen zo populaire begrip: *de lege verzameling*.

Indien men dus over een of meer artikelen van een oude aflevering wil beschikken, dan kan men kopieën trekken van de genoemde *sets* in M281, M377 of de tijdschriftenzaal van de Bibliotheek. (En verder bestaat er natuurlijk mijn persoonlijke geheime voorraad, waar ook kopieën van getrokken kunnen worden, zodra men zich heeft kunnen verzoenen met mijn hang naar corruptie en deze bovendien naar waarde heeft weten te schatten!)

We zijn overigens ook bezig te onderzoeken of, en zo ja hoe, we de betreffende informatie ook via het scherm van een workstation (of een PC, of een terminal) voor onze gebruikers oproepbaar kunnen maken. Daarover terzijner tijd meer.

Nu nog iets over toekomstige nummers. Het betreft dan de verdeling van de oplage. Ooit is er een lijstje gemaakt voor het aantal exemplaren dat er per afdeling nodig zou zijn. Het aantal personen in een afdeling fluctueert natuurlijk in de tijd en dus ook het benodigde aantal exemplaren. Misschien willen de secretaresses van de respectievelijke afdelingen ons berichten over een eventuele gewenste verandering in dat aantal te ontvangen exemplaren. Deze wensen kunnen naar de secretaresse van de afdeling CST, Mevr. Mieke Bruné, of naar ondergetekende gestuurd worden.

A.C. IJsselstein (acij@cwi.nl)



8. Enkele administratieve meldingen.

Hieronder volgen een aantal mededelingen. Misschien niet altijd even belangrijk, maar toch handig om te weten.

VERWIJDERING 'OUDE' FILES.

Sinds enige tijd, wordt 's nachts op alle Sun workstations met een lokale disk (SpareStation-1 en SparcStation-1+), gezocht naar files die weggegooid kunnen worden. Dit zijn met name *core* files en o.a. files die beginnen met ',', '# of '@'. Er wordt in /tmp, /usr/tmp en op de /ufs.private partitie naar deze files gezocht. Voorwaarde is wel dat de files minstens drie dagen niet meer gebruikt zijn. Dit gebeurde al op de *file servers*, maar is nu dus ook uitgebreid naar de Sun workstations. De SGI's volgen binnenkort.

/UFS.PRIVATE

Ik wil er hier nogmaals op wijzen dat het voor alle workstations met een lokale disk, mogelijk is om een eigen directory op de lokale schijf te laten aanmaken voor diegene die (bijna) dagelijks van de betrokken machine gebruik maakt. Hiermee komt niet alleen ongeveer 50 Megabyte schijfruimte extra ter beschikking(!), maar ook is men dan (bij een juiste configuratie) minder afhankelijk van *file servers*. Het blijft mogelijk om vanaf andere machines ook bij de files op de /ufs.private partitie te komen, en wel via de volgende padnaam:

/hosts/<machine-naam>/ufs/<gebruikersnaam>

.HOSTS

Een punt van voordurende zorg voor systeembeheerders is de beveiliging van de CWI machines tegen moedwillige indringers van buiten af. We zijn er echter van overtuigd dat een *open* omgeving zeer belangrijk is voor de op het CWI (en daarbuiten) werkzame onderzoekers. Een van de mogelijke 'gaten' hierbij is de zogenaamde *.hosts* file. Gebruik hiervan is zeer handig bij snel en makkelijk inloggen vanaf de ene CWI machine naar de andere CWI machine. We raden gebruik van deze file af, maar indien hier toch gebruik van wordt gemaakt, raden we sterk aan om het aantal entries tot het minimum te beperken (een of twee machinenamen is in 99.9%

procent van de gevallen voldoende!). Echter: informatie in deze file maakt het voor moedwillige inbrekers nog makkelijker om verder te komen en op andere machines in te breken. Om die reden is het belangrijk dat alleen de eigenaar van de betreffende *.rhosts* file deze kan lezen en schrijven. Om deze situatie zeker te stellen, draait er elke nacht een programma dat dan lees-en schrijfpermissies op deze file aanpast en lees- en schrijfpermissies voor anderen dan de eigenaar uit zet.

MAILBOX

Een aantal mensen gebruiken een mail programma, waarbij niet alleen de ongelezen mail, maar ook de gelezen in de *mailbox* blijft staan. Dit is op zich geen bezwaar, ware het niet dat de partitie waarop dit alles staat (i.e. */usr/spool/mail* op *charon*) begrensd in grootte is. Om in de toekomst problemen te voorkomen, raden we een ieder toch aan om mail in een folder binnen de *\$HOME* directory te bewaren. Hoe dit moet staat in de *manual page* van het gebruikte programma.

Frank Kuiper (frankk@cwi.nl)



9. De nieuwe situatie bij de SGI's

De Silicon Graphics machines op het CWI draaien verschillende versies van het operating systeem. In de laatste maanden zijn een aantal machines ook nogal eens van versie veranderd. In de volgende alinea's geef ik een kort overzicht van wat er nu draait op de diverse systemen.

Een aantal maanden geleden draaiden onze Silicon Graphics apparatuur de zelfde versie: Irix 3.3.2. We kregen toen echter de beschikking over een alpha release van Irix 4.0. Een *alpha release* van software is een release die de fabrikant wil testen. De gebruiker weet dat er allerlei fouten in kunnen zitten, dus hij mag niet klagen. Wel wordt hij geacht fouten te melden. Deze alpha release is op een paar machines geïnstalleerd, maar vanwege de te verwachten problemen is deze versie niet op alle machines geïnstalleerd. In de loop van de af-

gelopen maanden hebben wij een paar keer een nieuwe alpha release gekregen die op een beperkt aantal machines is geïnstalleerd. Uiteindelijk kwamen de Indigo's. Op de Indigo's zat een heel late versie van de alpha release. De eerste Indigo's die geïnstalleerd zijn (ongeveer 20 à 25) draaiden deze release. Onlangs hebben wij echter de definitieve release gekregen. Die release heet Irix 4.0.1. Deze is toen op alle 4D35's geïnstalleerd en op een aantal Indigo's. Ook *poseidon* en *guppie* draaien deze versie.

De situatie op dit moment is dat *meermin* nog Irix 3.3.3 draait (de server versie van 3.3.2), *poseidon*, *guppie*, alle 4D35's en een aantal Indigo's al Irix 4.0.1 draaien, en de meeste Indigo's een late Irix 4.0.

Sjoerd Mullender (*sjoerd@cwi.nl*)



