

STACKPOINTER

3-1989. Organ för Datorföreningen STACKEN, KTH.



STACKPOINTER

STACKPOINTER är husblad för datorföreningen STACKEN på KTH. STACKPOINTER sammanställs nästa gång 3-4 juni 1989.

Citera gärna. Men glöm inte ange källan. Tillskicka också föreningen ett exemplar.

Redaktör: Hans Nordström
 I redaktionen: Jan Michael Rynning
 Fotografer: Jan Lien, Peter Löthberg och Jan Michael Rynning
 Ansvarig utgivare: Mats O Jansson

Färdigställd: 1989-04-17

Datorföreningen STACKEN

Post till föreningen skickas till adressen eller läggs i postfacket på NADA.

Datorföreningen STACKEN
 c/o NADA
 KTH
 100 44 STOCKHOLM

Klubblokal: Lindstedtsvägen 5
 Datorhall: Brinellvägen 32, Teknikringen 22
 Telefon: 08-791 87 97

Postgiro 433 01 15-9
 Bankgiro 344-3595

Medlemsavgift 1989 är för studerande SEK 89, övriga SEK 189.

Ordf: Stellan Lagerström, stellanl@lne.kth.se, 08-790 78 14 (arb) 08-46 93 93 (hem)
 Sekr: Olle Betzén, betzen@lne.kth.se, 0758-318 48 (hem)
 Kassör: Henning Croona, hcroona@stacken.kth.se, 08-799 82 80 (arb) 08-739 17 40 (hem)
 Hexm: Johnny Eriksson, bygg@kth.se, 08-790 65 17 alt. 790 65 23 (arb) 08-27 24 30 (hem)
 Redax: Hans Nordström, sib@stacken.kth.se, 0760-405 69 (modem), 08-797 80 13 (arb)
 Övriga: Mats O Jansson, moj@stacken.kth.se, 08-34 93 20 (arb) 08-711 70 37 (hem)
 Thord Nilsson, thordn@stacken.kth.se, 08-712 00 90 (arb) 08-749 21 92 (hem)

Elpost: stacken@stacken.kth.se

UUCP alternativt: ...{uunet,mcvax,...}@stacken.kth.se!stacken
 ARPA alternativt: stacken%stacken.kth.se@uunet.uu.net
 BITNET/EARN alternativt: STACKEN@SESTAK
 SUNET DECnet alternativt: KICKI::STACKEN
 PSI alternativt: PSI%0240200101905::KICKI::STACKEN

I detta nummer

A-Data	4	Lysators 16-årsjubileum	14
Hur STACKEN fick datorkylare och -golv, samt fem stora omformare.		När Sveriges äldsta datorförening fyllde jämn 2-potens.	
DPS 8	9	Svenska Finans	18
Bankgirot	10	Hur STACKEN till slut även fick kylmedelskylare.	
Hur STACKEN fick resten av datorkylarna till sin nya hall, innertak och lite andra småsaker.		French Virus Alert	24
		GNU's Bulletin	25

NUCC-möte i Oslo

av Jan Michael Rynning



EN 16:e-18:e juni är det dags för en ny träff mellan datorföreningar vid universitet och högskolor runt om i Norden, nu för tredje året i rad. Den här gången är det Oslo Dataforening som står för arrange-

mangen. Förmodligen kommer STACKEN att ordna gemensam resa för dem som vill, precis som de två tidigare åren. Den som vill veta mer kan kontakta mig eller Erik Naggum, Oslo Dataforening, erik@naggum.uu.no.

Datorhallsbygget

av Jan Michael Rynning



ÖRDRAGARNA lade in sitt veto mot att vi lade in datorgolv innan de dragit färdigt rören. De ville ha ett stabilt golv att stå på, de-

ras jobb var svårt nog ändå. Men nu är rören i hallen färdigdragna, provtryckta och isolerade, så nu kan vi sätta igång med golvet.

A-Data

av Jan Lien

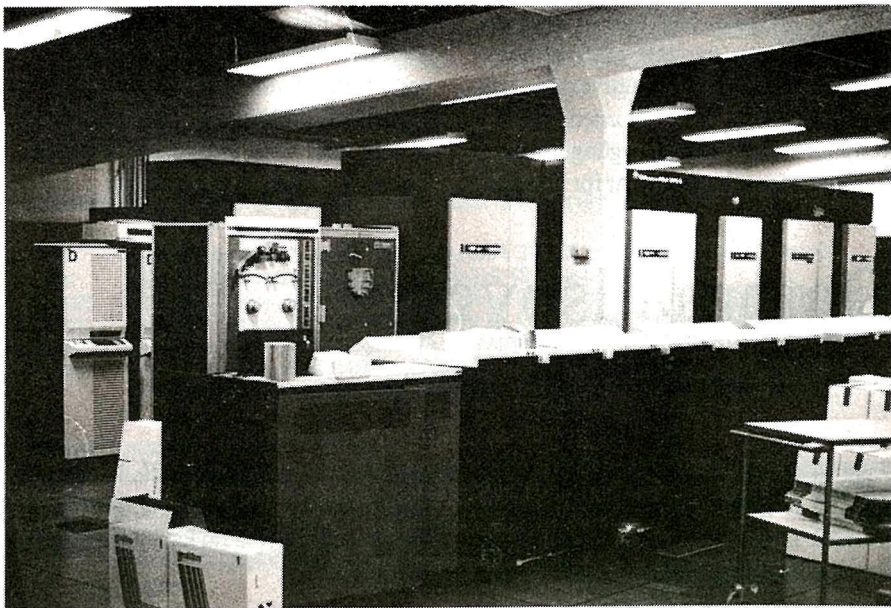


-DATA var en av de datorhallar jag sprang på när det gällde att tigga datakylutrustning. Tipset fick jag av en kille på en stor servicebyrå, som undrade om jag kände till att A-Data köpts upp av Conor och driften skulle slås samman med deras datorhall. A-Data var någon sorts administrativ datacentral, "Arbetarrörelsens Databehandling" ungefär och en ganska stor installation, såg vi. Deras maskinpark bestod av 3 st Bull DPS 8

med operativsystem GCOS 8 ("General Commercial Operating System 8"). De hade egen 10 kV-matning, dvs vid strömavbrott i Årsta var det bara A-Data som lyste vidare...

Vid första kontakten kunde inget bestämmas, de ville vänta med besked. Det jag fick besked om, var att vi egentligen hade kontakt med två företag, först och främst Conor, som ägde hallen och utrustningen minus själva

Två av A-Datas tre DPS 8-maskiner med kringutrustning.



datorena, samt Bull, som ägde datorena. Conor tipsade om att det som definitivt skulle kastas var omformare, som gjorde om 50 Hz ström till 60 Hz (amerikansk). Eftersom jag visste att amerikanska DEC-2020 går på 50 Hz ändå, men inte tillhörande diskar och inte heller KL-10:or, tyckte jag att det kunde vara en användbar pryl. Jag kontaktade därför Bull, som sade att jo visst, omformarna går bra men är en gammal modell, så en sån kan ni få. Därför åkte jag ut till A-Data för att titta och fick se 5 omformare, 4 i drift och en i reserv. De fanns i ett särskilt rum och förde rejält oväsen.

De roterande 60 Hz-omformarna.



Nåja, en sån skulle vi ha (nya omformare kostar pengar), kanske en extra som reserv. Vid närmare eftertanke, habegäret satte in, kanske det vore bra med alla. Nu hade vi fått 5 omformare av Bull. Jag tittade runt i datorhallen, IBM-mojuttan var redan flyttad, men de 3 Bull DPS 8:orna kördes fortfarande. Det var två äldre system, från 1980 och ett nyare, från 1984. De äldre systemen behövde 60 Hz, det nyare kunde köras på 50 Hz också om man ville. De äldre systemen hade imponerande skåp, minst 2 m höga, stora, men utan blinkande lampor. Programvarumässigt var skåpen utbytbara med varandra. När Peter fick veta att man hade DPS 8 undrade han om det var sådana man körde Multics på, en sådan maskin ville han ju ha. För den som inte vet, så är Bull DPS 8 36-bitarsmaskiner. Jag ringde Bull och fick efter lite betänketid veta att Multics körs visst på DPS 8, fast CPU:n heter då DPS 8M och är inte alls lika i bakplanet. Peter tyckte då att "kan de bara skaffa fram ritningar på vad som skall byggas om så tar vi burken och bygger om den". Bulls tekniker tyckte inte som Peter, det fanns för det första ingen ritning på ändringarna, för det andra hade ingen gjort det. Dessutom hade de redan sålt den nya DPS 8:an till utlandet, de gamla systemen skulle bli reservdelar. Diskussionen slutade med att vi fick ett komplett gammalt system minus CPU (som vi ändå inte kunde köra Multics på) och minus primärminne, som blev reservdelar. Så nu behöver vi bara im-

portera två skåp, egentligen. Tyvärr fick vi inga 1 GB diskdrivar, de var alla sålda med det nya systemet. Det här blev klart en tid innan systemen skulle stängas. När det gällde den ytterst intressanta kylsidan, visade det sig att Conor redan hade lagt ut rivningen av datorhallen på Enaco. Det låg även mycket IBM-kabel kvar under golvet. Den visade sig tillhöra IBM, så jag ringde dit och vi blev raskt lovade att all IBM-kabel som finns kvar 2 dagar efter stängningen kunde vi ta, då hade IBM tagit vad de absolut skulle ha.

Kyla fanns det ganska gott om, jag tror att det fanns 4 st Lidingö 50 kW, 3 st Liskey 45 kW, en Hiross 40 kW, 2 små Liskey 25 kW. På taket fanns tre stora Asarum kylmedelskylare, ett par tre mindre kylmedelskylare, samt några små freonkondensorer. I källaren fanns en stor Westinghouse kallvattenmaskin, som tydligen aldrig hade fungerat riktigt bra. Den funderade vi ett tag på att ta med, men då skulle man behövt montera isär den i dess atomer, den var nämligen ihopskruvad på plats. Jag fick kontakt med en kille på Enaco att tigga kyla av, han hade en positiv inställning. Peter tyckte att han hade kontakter på Enaco, så det slutade med att jag, Peter och jag tror även Hans Nordström, ägnade oss åt att förhandla med Enaco. Det visade sig aningen komplicerat, eftersom Enaco redan hade sålt alla kylmaskiner och tillhör där till en kylfirma, Hansson kyl.

Så kom då dagen S, S som i *stäng* datacentralen. Det vi hade fått löfte om var just bara Bull-prylarna. Efter att ha blivit väckt per telefon, gav jag mig iväg för att arkivera vår Bull-maskin. Det visade sig att Bull hade 3 st tekniker där som monterade isär maskinerna, de hade var sin uppgift. Alla tre teknikerna hittade i en oanad takt prylar som vi behövde och de kastade kablar och muttrar åt mig under glada tillrop som: — Ta hand om 5 av de där kablarna som går i härva under golvet, dem behöver ni!

— Ta vara på det här skåpet, det är ert!

— Kom ihåg vilka kablar som gick in här och vilket skåp de gick till, de får inte kopplas fel!

— Ta vara på kablarna innan någon klipper av dem!

Jag kände en lätt känsla av otillräcklighet. Efterhand som dagen gick kom lite mer folk, men faktum är fortfarande att Bull-teknikerna egentligen packade prylarna åt oss.

Det fanns fler personer än Bull-teknikerna där på morgonen. Jag mötte representanten från Enaco, som sade att de själva hade så mycket att göra att de hade lagt ut rivningsjobbet på en underleverantör, som jag också mötte, en lång kille med glasögon och getskägg. Där fanns också ett par killar från Conor, som räddade det sista undan rivningen. Jag och Conor-killen gick runt och tittade på anläggningen, vad som fanns kvar och vad vi kunde ta. Vi hittade bandställ i olika former, rullbara ställ, ställ i hyllform, hängställ i hyll-

form, samt ett sånt här riktigt arkiv, ett sånt med stora skåp som går på skenor i golvet. Alla dessa fick vi ta, vi lämnade hängställerna och det riktiga arkivet på skenor. Vi hittade transportväskor för band, dem tog vi och vi besiktigade ett antal modemskaåp, modell äldre och tunga, som vi fick *alla* av. Jodå, *alla*, det var vid kl 11.00 ungefär.

Dagen efter fick i alla fall Peter löfte om att vi fick ta 2 st Lidingö-kylar. De såg snygga, nya och välbehållna ut. Någon av oss var effektiv och satte snabbt upp en klisterlapp med texten "STACKEN" på de två kylar vi ville ha. Genast var någon annan där och skrev kommentaren "Vem har sagt det?" på lappen. Det visade sig att vi hade problem med Enacos underleverantör, killen med skägget. Han hade nämligen siktat in sig på allt i hallen (jodå, det är sant, vi har mött en större samlare än Peter), han kunde nämligen använda eller sälja rubbet. Tex så monterade han ned alla innertaksplattor och plattor i väggarna för att använda som isolering i sitt lager, datorgolvet skulle han använda som golv i samma lager, modemskaåpen påstod han sig kunna sälja för 5000 kr styck (vad kostar nya egentligen?), ... Nåja, efter lite diskussion redde saken ut sig, förmodligen fick han veta av Enaco att han ändå inte kunnat ta någon kylmaskin själv. Han var faktiskt på gott humör ett par gånger, både JMR och jag fick hans visitkort vid skilda tillfällen, när han försökte övertala oss att göra så mycket som möjligt av hans jobb i utbyte

mot några innertakslister eller lite annat skräp.

Nåväl, STACKEN satte igång att demontera våra omformare, som visade sig väga närmare 1,5 ton styck. Genom att montera isär dem i delar fick man en motor och en generator, som vardera vägde en del över 500 kg. Det var ganska trångt inne i omformarrummet och man fick lyfta upp delarna med en motorblockslyft, som inte var speciellt lämpad för uppgiften egentligen. Hydraulpumpen läckte dessutom, så när man hade kommit över en gräns fanns ingen återvändo, antingen bar lyftet eller så rasade generatorm ner på fel ställe. Det gick trots allt ganska bra. Efter en stund stod generatorm på pall, pallen stod på pallyften, som i sin tur stod på en bädd av datagolv ovanpå stativet som egentligen allt stod på från början. Nu var allt som återstod en backe att köra pallen på ner till golvet. Den skapade man genom att lägga golvplattor i trappa, vadå jämn lutning, pallyften fick sån bra fart ändå, det sade bara studs-studs-studs-klofs så var den nere.

För att få ut kylarna fick vi låna ett par mojutter av Enaco, som skulle visa sig revolutionera STACKENS kylfraktavdelning. Det var två lyftvagnar, som kunde användas genom att spännas runt objektet. Med de vagnarna gick lyften oerhört smärtfritt, sedan kunde man ganska lätt rulla ut kylarna. Ett mindre problem uppstod i hissen, det visade sig att kylarna var längre än den.

Genom att ta av kortsidorna på kylarna fick man precis in dem i hissen och kunde få ned dem, trots allt. Lidingö-kylarna är av någon anledning lägre än alla andra datorhallskylar vi sett, så de gick faktiskt att pussla in i en lastbil med skåp. Däremot kunde man inte få upp kylskåpen på lastbilens bakgavelhiss, den var för liten. Nu visade det sig att man hade en pallyft modell kombi-super där, en sådan som ser ut som en korsning mellan pallyft och gaffeltruck. Man använder den för att lyfta upp sina pallar på hyllor, när man inte har behov av en riktig gaffeltruck. Någon (Peter?) kom på att man kunde köra upp ena änden av ett kylskåp på baklyften, med pall och pallyft under. Sedan lyfter man helt enkelt baklyften långsamt och och pumpar upp super-pallyften i samma takt utanför baklyften. Enda problemet var när man lastade ur den andra kylan, då "råkade" super-pallyften hamna längst in, där man inte fick ut den. Vissa i föreningen hävdade att ingen skulle märka att super-pallyften saknades, andra var av motsatt åsikt och till slut återvanns super-pallyften med stor arbetsinsats.

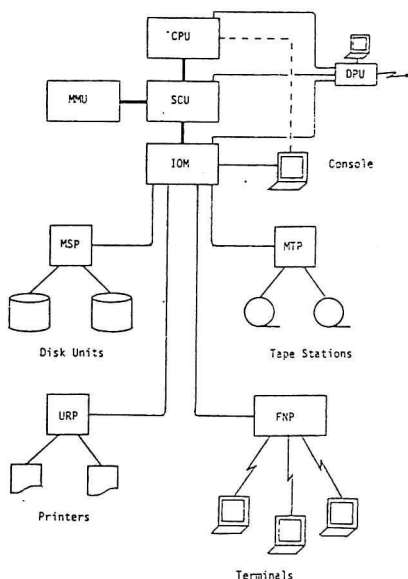
Nu hade STACKEN fyllt på sitt lager rejält, en Bull-maskin, 5 omformare med tillhörande elskåp, 2 kylmaskiner, ett par pallar elkabel och lite annat skräp. Det var inte slut här, vi kom att få gott om datagolv också. Det arbetet blev jag avkopplad från. Vi hade nämligen dagligen diskussioner med killen med skägget om vad vi kunde ta och en dag blev han så jobbig att han ville kasta ut oss direkt, varpå jag tog mig ton och sade att vi faktiskt hade fått modemskaåpen innan han fått dem (en annan kille från Conor hade gått runt med skägget några timma senare än med mig och lovat honom samma prylar som vi redan fått). När jag då påpekade hur mycket IBM-kabel som låg under golvet och hur många tusen det kunde kosta honom när vi istället köpte ny kabel om vi inte fick plocka den, ja då blev jag så nära lynchad som jag någonsin varit och sedan fick jag strikta order att inte visa mig mer i omgivningen. Ordern kom från STACKEN! Nåja, det var ju ganska skönt att kunna ta igen sig utan dåligt samvete! Till slut hade i alla fall de andra arbetat ihop riktigt mycket datalov, duktigt gjort!

DPS 8

av Jan Lien

BULL DPS 8 är ett stort och kraftfullt system för administrativ databehandling, med mycket disk och många terminaler. (STACKEN har fått en DPS 8, se artikeln om A-Data).

Systemen kan kopplas ihop till tandsystem. Ett typiskt DPS 8-system består av flera skåp, varav tre stora skåp utgör den centrala processorn (se figur).



De tre skåpen är:

- CPU (Central Processing Unit)

- SCU (System Control Unit) med MMU (Memory Management Unit) inuti

- IOM (Input/Output Multiplexor)

Alla dessa skåp pratar med DPU (Diagnostic Processor Unit) som i sin tur pratar med konsolterminalen. Inga andra terminaler pratar direkt med dessa tre skåp.

Förutom de stora skåpen, finns ett antal mindre skåp, som en köksbänk med ett litet spisreglage på. De är:

- MSP (Mass Storage Processor) för diskar
- MTP (Magnetic Tape Processor) för bandstationer
- FNP (Front-End Processor) för terminaler
- URP (Unit Record Processor) för printers

Vi har inget separat URP-skåp. Vår URP är inbyggd i IOM-skåpet. På DPS 8 använder man huvudsakligen synkrona Bull-terminaler.

Det operativsystem man körde var GCOS 8. Det finns ett annat man kan köra på den modellen, lär vara mer användarvänligt, heter CPS 6. Saxat ur beskrivningen av GCOS 8 i Bulls datablad:

- Up to 64 MB of memory.

- Common file system with comprehensive file protection and access control.
- Powerful multifunction Time Sharing System capable of handling as many as 600 time sharing users at one time.
- Extensive software library to support all facets of business, educational, and governmental operations.
- 1 MB program size in single segment mode
- 16 MB in multi-segment mode.
- Up to 240 files per activity.
- Compilers: COBOL-74, COBOL-68, FORTRAN-77, FORTRAN-IV, PL/I, RPG-II, BASIC, PASCAL, APL/66, LISP, B-LANGUAGE, C, GMAP macro assembler.
- Data base system: Data manager, Transaction processor, Query and reporting processor, Procedural language interface, Time sharing interface.

DPS 8 använder antingen 9-bits ASCII (fyra tecken per maskinord), BCD (Binary Coded Decimal) 6-bit med 6 tecken per maskinord eller Packed Decimal, 8 st fyrabitars tecken per maskinord.

Bankgirot

av Jan Lien



ANKGIROT var en av de första hallar som nämndes när insamling av kylutrustning kom på tal, alla verkade veta om den. När jag hade ringt runt till många andra hallar, tyckte jag nog att det fanns skäl att ringa Bankgirot ändå. Det visade sig att det inte var någon annan som gjort det. På Bankgirot ville man vänta och se vad som skulle hända, det var inte helt klart vem som ägde kylarna, om det fortfarande var Bankgirot eller om det var byggherren, Skanska. Efter lite vidare diskussion kunde jag i alla fall åka dit och titta på vad som fanns och säga hur mycket vi behövde av

det. Jag räknade in en massa kylar som skulle ut, 9 st Hiross, 3 Liskey och flera kylmedelskylare. Kylmedelskylarna brukar stå svårtillgängliga på tak och behöva lyftas ned med stor kranbil eller med helikopter. De här kylmedelskylarna stod på taket och man kunde lätt köra upp på taket med en vanlig kranbil, dvs fantastiskt lätt att komma åt. Enligt uppgifterna vid inspektionen kunde vi även ta ett antal kylmedelskylare. Sedan gick dagarna och inget hände.

Så kom dagen Ö, Ö som i *överlämnande* av nycklar, dvs Bankgirot skulle

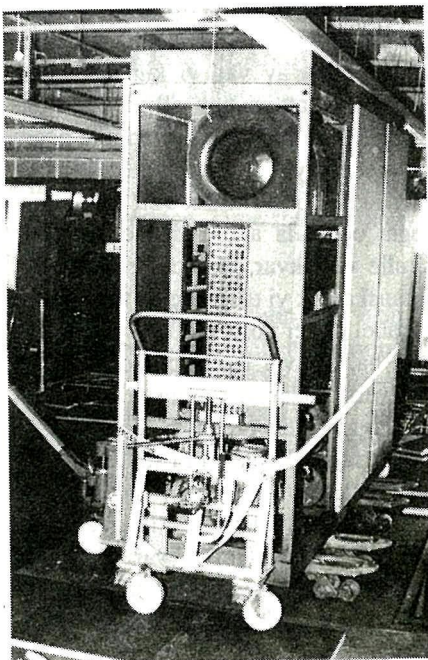
lämna över lokalerna till Skanska, som skulle börja ombyggnaden. Nu klarade bilden av vad vi kunde ta. Det visade sig att vi fick alla kylar i datorhallar av Bankgirot, men inte några kylmedelskylare. De kylmedelskylare som skulle bort tillhörde ett leasat aggregat och alla andra kylmedelskylare skulle vara kvar, för kontorskylan. Av Skanska fick vi tillåtelse att ta golv och elkabel under golv. Golvet ville vi nog inte ha, det var tunna plattor, med ett ramverk under, dvs ganska besvärligt att lägga. Den typen av golv är inte heller så stark, på andra ställen har jag sett dataskåp gå igenom sådana golvplattor med hjulen, så att säga sjunka genom jorden. Golvet var för övrigt schackbrädesfärgat, gröna och blåa rutor. Vi fick tydligen nog med golv på A-Data.

Nu gällde det först att få loss kylarna. En massa medlemmar engagerade sig med att skruva loss rör med glykolvatten. Till slut var kylarna bortkopplade och frågan transport vidtog. Man skulle ju behöva en kranbil för att lyfta på och av dem, det fanns ingen stor lämplig lastramp. Jag ringde runt och försökte hyra kranbil, men se det gick inte, de får man bara med chaufför, betalas per timme tack. Då började jag ringa lastbilsförsäljarna, för att höra om de hade någon bil att låna ut. VAG ute i Hammarby, som säljer Volkswagen, ställde upp. "Jo, vi har en kranbil, den kan ni väl få låna", svarade man och jag beskrev vad vi skulle lyfta, $2 \times 1 \times 2,2$ m kylar som väger 900 kg. Så kom hel-

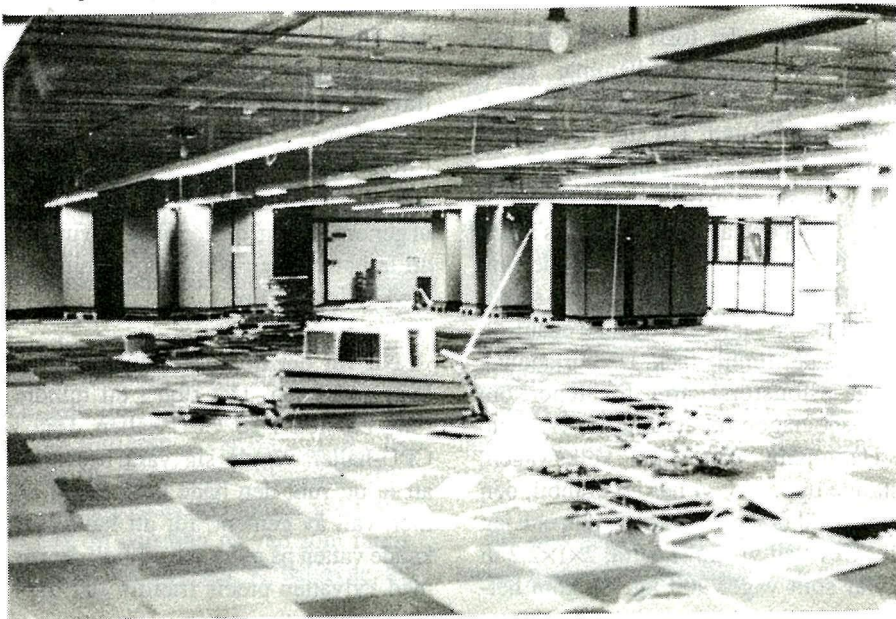
gen när transporten skulle ske, jag åkte ut till VAG fredag eftermiddag för att hämta lastbilen med kran. När jag kommer fram, går vi ut och tittar på kranen. Det visar sig vara en liten kran, som lyfter max 750 kg 1,5 m över flaket... Mummel, fredag eftermiddag, för sent att hitta på något annat, kylarna går ändå inte in i en vanlig skåpbil (fri höjd 1,95 m mot kylarnas 2,2 m). Det var inte mycket annat att göra än att åka till Bankgirot med bilen. Där hade samlats ett stort antal medlemmar, till och med Peter Löthberg var med för att flytta kylarna. Gissa vad Peter sade när han såg "kranbilen". Han höll på att säga saker om bilen i över en timme, sedan konstaterade han att kylarna skulle flyttas — nu. Tack vare att lastbilen var så liten, gick det bra att backa upp den mot dörren. Med mankraft sköt man upp en kyl på det lilla flaket, spände fast den och körde till KTH, där man backade in i höglagret och lyfte av kylan med travers. Proceduren upprepades för alla nio kylarna. Till slut hade alla kylarna i övre hallen flyttas ut, det gick på en natt. Kylarna var så höga att de inte gick ut genom dörren, därför blev vi tvungna att ta bort dörren och höja öppningen, med hjälp av kofot.

Nu var det tre Liskey-kylmaskiner kvar i källaren. Två stod i ett mindre rum, en stod i fd CPU-hallen. Den i CPU-hallen visade sig vara besvärlig att få ut, runt den hade man byggt en liten vall av cement, för att hålla läckande vatten på plats och så gick det rör med kylvatten precis framför där man

Med hjälp av bräckjärn lyftes datorkylarna några cm, så vi fick in gafflarna på lyftvagnarna. Efter att ha spänt fast dem och pumpat upp domkrafterna, kunde vi rulla ut kylarna på helt golv. Sedan vidtog arbetet med att flytta över dem på pall. Två pallyftar sköts in och pumpades upp, lyftvagnarna togs bort och högar med golvplattor sköts in under hörnen på kylarna. Efter att i ett par omgångar ha sänkt ner kylarna på platthögar, lagt golvplattor på pallyftarna, höjt igen och lagt fler plattor i hörnen kunde vi skjuta in pallar under kylarna.



Datorkylarna i övre hallen uppställda i väntan på borttransport.



ville köra in pallyften under den. Golvet i den hallen hade nämligen Bankgirot tagit med sig. Första dagen när vi kom sade de att "ni bör nog börja i nedre hallen, för där tar vi ut golvet i morgon". Och dagen efter var golvet verkligen väck. Bara några plattor var kvar runt varje kyl. Det slutade med att vi lämnade kylan i CPU-hallen och bara tog de två maskinerna i lilla rummet. Dem fick vi ut genom att hissa upp dem med pallyftar, lägga under en massa golvplattor och gradvis sänka. Till slut var de så långt ned att man kunde använda lyftvagnarna och rulla ut kylarna i gången och fram till hissen. Senare lät vi en kranbil hämta dem och köra över dem till höglagret. De kom att hamna utanför höglagret, eftersom jag inte hade nyckel dit in. För att få in dem gick jag upp till Provvningsanstalten bakom, frågade om vi kunde få låna deras gaffeltruck en dag och jo se det gick bra. Den har vi lånat vid flera tillfällen senare.

Under arbetets gång hade en del plockat kabel, både kraftkabel och vanlig seriekabel. Arbetsledaren från Skanska hade sagt att vi kunde ta lysrörsarmaturer. Det var gula metallarmaturer som faktiskt var riktigt snygga med tanke på att det ju egentligen var en industri. Så under en del arbete monterades 4 pallar (= 160 st) lysrörsarmaturer ned. Nu hade vi kylar i över-skott, lysrörsarmaturer, en hel del kabel, ja vad var det då egentligen som saknades? Jo, vi saknade kylmedelskylare och innertak, av de stora sakerna.

Peter tyckte att vi borde försöka få innertak, så jag pratade med platschefen hos Skanska. Han undersökte saken lite och sade att, "Jo, visst kan ni ta innertak". Vi hittade en arbetsvagn som gick att köra runt i lämplig höjd. MOJ och Henning fick uppdraget att klättra upp och plocka ned plattor, som vi andra tog emot. Det var bara att köra runt vagnen när ett område var avverkat. På så sätt tog vi med innertak och upphängningslister till hela det tak vi kunde nå från vagnen. Lite tak blev kvar, eftersom golvet var helt uppbrutet på vissa ställen, och man följaktligen inte kunde köra vagnen dit. När vi hade plockat ned innertaket, kom killen från Skanska tillbaka och sade att det blivit lite fel, innertaket skulle varit kvar... "men det gör inget", tyckte han ändå!

Vissa andra saker följde helt enkelt med, tex fanns två trevliga elskåp med ampèremeter, mätområde 0-300 A, som passade oss utmärkt. Några saker flyttade vi på, utan att ta med dem, ett exempel är elskåpet som föll i gomet med ett öronbedövande brak, efter att en våra medlemmar börjat studera kofotens möjligheter. Han bände och slet på ett helt annat ställe när skåpet "tappade" balansen. Varför blir alla så destruktiva med en kofot i handen?

Som avslutning kan sägas att Bankgirot var ett av de ställen som gav oss mest användbara prylar, en riktig höjdarhall med andra ord. Kylmedelskylarna fixade vi senare, på Svenska Fians.

Lysators 16-årsjubileum

av Peter Löthberg



Å Lysator hade annoserat ut att de skulle ha 16-årsjubileum och även sökte föredragshållare, bestämde jag mig för att åka dit.

Festligheterna avlöpte enligt normala rutiner. Lysator hade, mot all förmodan, lyckats få upp sin av skolan utlånade DEC-2020, Elinor. De hade fått låna diagnostiksystemet av DEC, och det snurrade bevisligen när jag kom dit.

Jag hade taget med en bootbar RP06, med ett TOPS-20 V4.1-system och DECnet fas IV-kod hopskruvat. Som vanligt hade jag ju missat att sätta de flesta parametrarna rätt, samt glömt en vettig SETSPD. Men det finns ju FILDDT.

Med hjälp av Larsa och Bo Kleve från Lidac, som lånnade ut två basbandsmodem till Lysator, kopplades maskinen upp på DECnet/Sunet under natten. Sen dess har den faktiskt fungerat ganska bra vad jag sett.

Alla STACKEN-medlemmar har konto på den, de heter "STACKEN.riktigt-namn", passerord kan man få från mig.

Elinor kör även en fräsch version av MM — till skillnad från de andra DEC-20-maskinerna i Linköping — den vet att den heter *elinor.lysator.liu.se* och talar post med SMTP över DECnet. Gateways mot TCP/IP utgör RTR59B och Kicki.

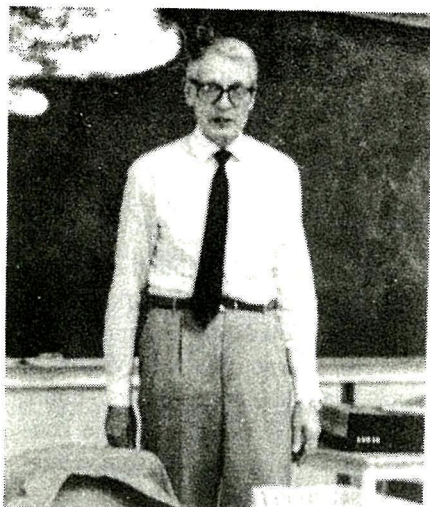
Jag får gratulera Lysator till den uppräckning som skett och hoppas att de kommer att få stor glädje av sina maskiner. (Man börjar fundera på om det inte är så att Lars Aronsson, under sina månader i Stockholm lärde sig något...)

På sista raden skall jag faktiskt utdela ett visst beröm. På söndagen efter festligheterna var jag kvar och fixade med Elinors TU45 och drabbades av akut hunger. Och vips hade Lysator trollat fram en riktig frukost. Det klarar inte STACKEN idag.



Lysator med gäster samlade i ISYs seminarierum i väntan på föredrag.

Viggo Wentzel, tidigare hårdvaruchef på DataSAAB.



Sven Yngvell, tidigare systemprogrammerare på DataSAAB.



Lysators 16-årsjubileum

av Lars Aronsson, Lysator

Fredag.

19.00 Pell får upp Elinor. Hurra! Eftersom någon (DEC?) äntligen släppt till en bootbar packe. Läs mer i mötet "hackers (@) elinor". Mer förberedelser under kvällen.

hysteres på bakelserna, brygg kaffe, breda mackor, skriva skyltar, hänga upp serpentiner (fanns ej ballonger i Ryds Vivo).

10.00 Första gästen Olov Giertz kommer. Pul-Ess har kommit.

Lördag.

03.00 Jag somnar alldeles för sent.

10.45 Peter Löthberg anländer och överlämnar en bootbar packe med 4.2-monitor med DECnet fas IV till Elinor.

07.00 Jag vaknar.

11.00 Officiellt start för drop-in. Folk droppar in. T-J Hallberg, Sven Yngvell, Viggo Wentzel, Bo Ragnemalm, Lennart Pettersson, Lennart Löfgren, Jan-Erik Stjernvall, Bo Kleve, Lars-Anders Eriksson, Jan Danielsson, Mats Lenngren, Göran Åhlström. Peter Löthberg och Ulla någonting. Jag, Pul-Ess, JSK, Henrik, Magnus Redin, Olov Giertz, Kjell Enblom och massa fler som jag inte orkar räkna upp. Folket äter yoghurt, kaffe, te, skorpar, mackor. Vi saknar Kjell Ståhlbom, Anders Thulin, Göran Karlsson som inte kunde komma och flera som vi missat att bjuda. Folk lyssnar på ett

08.20 Jag är på skolan och kommer inte in, för jag är först och har ingen nyckel. Lennart Ljung hemma och trivs.

08.30 Magnus Redin kommer från Åtvid via bageriet där han hämtat dagens succé, dvs lysatorbakelserna.

08.40 Keijo Hilden kommer till skolan och öppnar dörren åt oss. Vi går in och börjar förbereda.

09.00 Pell kommer. Läser upp fikarum o dyl. JSK och Henrik kommer. Förberedelserna tar fart. Pudra

band med datamusik från D21 och D22.

- 11.45 En delegation från Ctrl-C dyker upp och överlämnar en present. En exor-mac-terminal med flyttade tangentskivor och extra ditlimmade. Inga tips om hur den ser ut. Den finns i Lysators lokal att beskåda.
- 12.00 I stället för att börja, går vi ner i lokalen. Kjell Enblom anländer med bandspelare för att kunna spela in föredragen.
- 12.30 Vi börjar. ISYs seminarierum. Jag hälsar välkommen och drar något snabbt om Lysators start 1972/73. Viggo Wentzel håller föredrag om svensk datorhistoria från 1943 (ja!) till 1960. Sven Yngvell håller föredrag om SAABs räkneautomat, SARA, samt spelar ett band med datamusik från BESK. Snacka om pionjäranda!
- 14.00 Kort paus. Kaffeskvätt.
- 14.15 Robert Forchheimer håller föredrag om LYS-16. Vilka tider!
- 15.30 Längre kaffepaus. Lysatorbakelser. Succé.
- 16.00 Peter Löthberg håller föredrag om SUNET, NORDUnet, STACKEN, STACKENS nya lokal samt
- NUCC. Alla fyra fördragen finns på band, som kanske behöver redigeras något.
- 17.00 Visning av ISYs 3D datorgrafik på PICAP-3. Mer visning av lokalen. Installation av LiDAC lånade 19k2-modem mellan Elinor och Clover för DECnet. Köp av tråd av Admittansen och provisorisk dragning.
- 19.00 Uppsamling för färd mot Herrgård.
- 19.30 Ryds Herrgård. Middag i 3 rätter.
- 23.00 Alla har fått kaffe och pratstund. Få stannar på disco. Långväga åker hem. Peter Löthberg och många lysiter tar sig tillbaks till skolan för att få fart på DECnet.
- 02.00 Varför åka hem när man kan stanna och se på video. TRON fanns inte, men ett par andra sopigare SF-filmer dög.
- 05.00 Varför somnar jag så sent när jag skall upp dagen efter?

Söndag.

Jag kunde inte vara med under söndagen, men rykten säger att det var fullt liv.

Svenska Finans

av Jan Lien



ET finns alla möjliga sätt att hålla sig informerad om datorhallar som skall rivas — ett är att läsa dagspressen. Jag såg nämligen en annons i Dagens Nyheter om "Datorhall uthyres komplett" och ringde givetvis för att höra vad det var. Det visade sig att det var ett kreditföretag, Svenska Finans, som hade sin datorhall på Hälsingegatan och nu skulle flytta datorn till lokaler tvärs över gatan, för att samla alla sina saker under ett tak. Däremot skulle de inte ta med sig kylanläggning och datorgolv, så jag framförde att vi ju var intresserade av allt som kunde bli över, framför allt kyla. Svaret blev, att ja, det kan vi ju kanske tänka oss att byta prylar mot att ni tar ut så mycket som möjligt, så slipper vi betala för rivning, det vill säga om vi inte kan hyra ut. Lite senare när jag hörde av mig igen såg det mörkt ut med uthyrningen, så vi kom överens om att jag skulle komma och titta på vad som fanns där egentligen. Det visade sig att de hade en IBM 3090 i en ganska liten datahall (mycket mindre än vår nya), med ett lustigt golv, 75×75 cm plattor, skruvade i ramverket, på låga stöttor. Dessutom i efterhand klädda med heltäckande matta, skuren efter golvplattorna i en del rum. I kontorsrummen låg samma datagolv,

fast där var det klätt med heltäckande matta som inte alls var skuren, dvs jag förstår inte hur man skulle kunna lyfta på några av plattorna. Jag konstaterade raskt att vi inte alls var intresserade av golvet. De hade kallvattenkylning i hallen, dvs de hade två kylaggregat som tillverkade kallvatten, kallvattnet gick sedan i datorhallen, där man hade 8 st Holland Heating fläktar som blåste ut kylan i luften. Det fanns två kallvattenmaskinerna, en hiskelig 4×1,5×1,5 m RC Condizinatori, som stod helt inomhus och sög in luft i ena änden genom ett spjäll, körde luften genom kylbatteriet och viftade ut den nu varma luften genom ett annat spjäll. Verkligen en annorlunda anläggning. Den andra maskinen var en Bitzer okänd modell, som hade monterats in senare. Den såg ganska ny och fräsch ut, två kompressorer och en takmonterad Asarum återkylare, modell stor. Så synd, vi hade ju redan projekterat kylning med varmvatten, dvs vi har ju kylskåp där varje skåp är en egen kylproducent och där man ventilerar bort värmen via varmvattenledningarna. Därför kunde vi inte på något sätt använda de blå blås-skåpen i datorhallen. Möjligen skulle vi kunna använda RC-historien till att kyla av vattnet från hallen. Nåja, bättre att ha något än inget, så jag pax-

ade lystet för allt med kyla i. Nu kunde jag ju inte anta att vi kunde använda något, så jag ringde närmaste vänliga kylmontör och frågade om de ville byta in något. Svaret var att jodå, alla skåpen, framförallt blåsskåpen, var intressanta som paket. Det kunde ge en del pengar, som vi skulle kunna använda till att köpa rätt sorts kylmedelskylare. Vi hade ju vid den tidpunkten gott om datorkylare, men inget att kyla med på utsidan.

Efter ytterligare några månader kom dagen S, S som i *stängd* hall. Svenska Finans skulle använda helgen till att flytta sin dator, så vi kom överens om att jag skulle komma på måndag morgon och titta på situationen. När jag kom på måndag-morgonen fick jag klart för mig att här skulle det gå fort, hela hallen var rensad, inga kylar fanns kvar i hallen. Det visade sig att de hade fått hjälp av KLIAB att montera in sin nya kyla och ut den gamla. Som tack tog KLIAB hand om alla blåsskåpen i datorhallen, de som jag riktat in mig på. Efter lite palavrerande med Svenska Finans kom vi överens om en bra lösning för alla parter, framförallt för STACKEN. Överenskommelsen var att STACKEN tog ut de två stora kallvattenmaskinerna och som tack betalade Svenska Finans nedtransport av Asarum kylmedelskylaren på taket (den skulle ju bort ändå). Det visade sig att det bästa och billigaste sättet att få ned kylmedelskylaren var att lyfta ned den med helikopter. Det kostade ungefär 11.000 kr. En oväntad men absolut inte

önskad effekt var att KLIAB hade tömt hela systemet på glykolvatten, så vi slapp det problemet. Som ni känner till får man ju inte tömma ut glykolvatten var som helst (om man inte är flygplats förstås).

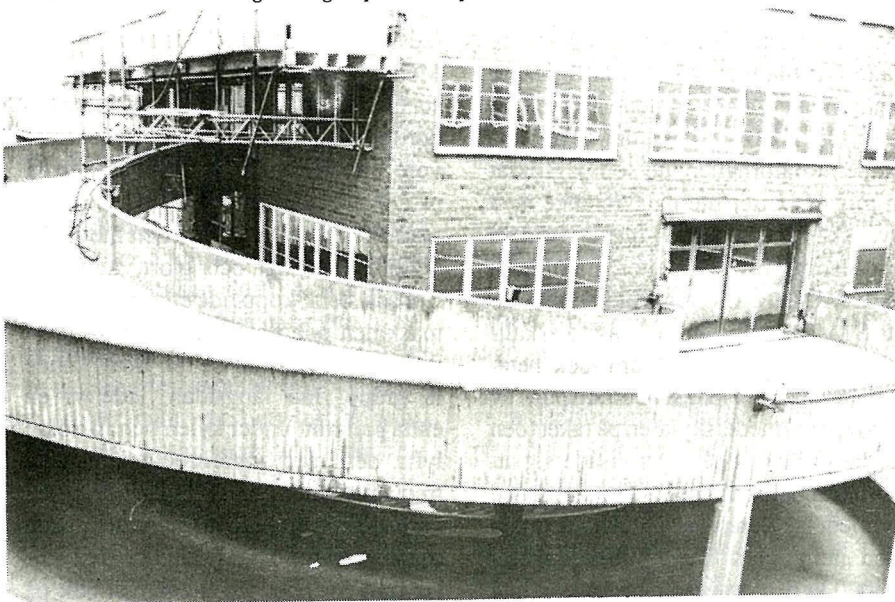
Jag och JMR satte igång att skruva loss alla rör och pumpar. Under tiden dök en kille från Svenska Finans upp, Leif och konstaterade att det stod en gipsvägg i vägen för att få ut kylarna. Han tyckte att det kanske vore något att riva med gaffeltrucken han hade med sig. Ja, vi motsatte oss inte det och i ett moln av gipsdamm försvann väggen. Jag och JMR räddade snabbt våra kläder undan gipsdammsmolnet till ett annat rum. Sedan gick han lös på nästa vägg. Det gick snabbt och effektivt att få bort väggarna, fast vi kan konstatera att det finns effektivare redskap än gafflar för att riva väggar. Med väggarna borta kunde vi konstatera att RC-maskinen var ju redan bortkopplad från rörsystemet. Därför vore det ju smart om man kunde få ut den lite mer mitt på golvet, så att man lättare kan attackera den med lyfthjälpmedel. Sagt och gjort, Leif, vår nye assistent, kopplade ett rep runt RC-maskinen och gaffeltruck och drog ut den. När den var ute så man kunde få grepp från långsidan, lade vi en enda pall mitt under kylen och försökte lyfta den med gaffeltrucken. Gaffeltrucken skulle lyfta 1000 kg och lyften verkade klara tyngden, däremot tippade trucken. Fast med JMR och två killar från Svenska Finans längst bak på

trucken som motvikt så gick det att lyfta och köra runt kylen. På så sätt fick vi ut den mitt på golvet. Det gick bra mycket fortare tack vare den hjälp vi fick, och Leif med gaffeltrucken var nöjd också, riva väggar var verkligen ett nöje, tyckte han! Varför blir alla så nöjda av att förstöra, parallellfall är folk med kofot?

Senare på kvällen dök Stellan, Henning, Thord m fl upp och vi kopplade isär alla rör till den återstående kylen. Vi tog vara på ventiler och termometrar, lite kabel som ändå låg lös ovanpå all annan bråte. På en dag och kväll, tisdagen, hade vi fått loss alla kylar, kopplat bort rör och ventiler, kort sagt allt var färdigt att hämta. Ovanligt snabbt arbetat, säkert vårt hittills snabbaste och effektivaste hämtningsjobb.

På torsdagen beslöt någon (JMR?) att kylarna skulle ned på gatan, så i god tid hyrde jag en lite lastbil, lånade in ett par pallyftar från Stockholms MFO som vanligt, så hade Svenska Finans en pallyft och så tog vi med Peters lilla pallyft. Den enda vägen ner var nämligen längs en ramp i tre våningar, det hade förut varit bilfirma i lokalerna. Rampen var så låg att man inte kunde köra upp en lastbil. Idén var att sätta kylen på två eller fyra pallar, spänna fast pallarna med spännband så att inte kylen ramlade av pallarna och sedan köra ekipaget nedför rampen. Vi valde till slut att använda fyra pallar och fyra pallyftar, eftersom det gav bättre stabilitet.

En del av den tre våningar höga spiralrampen.

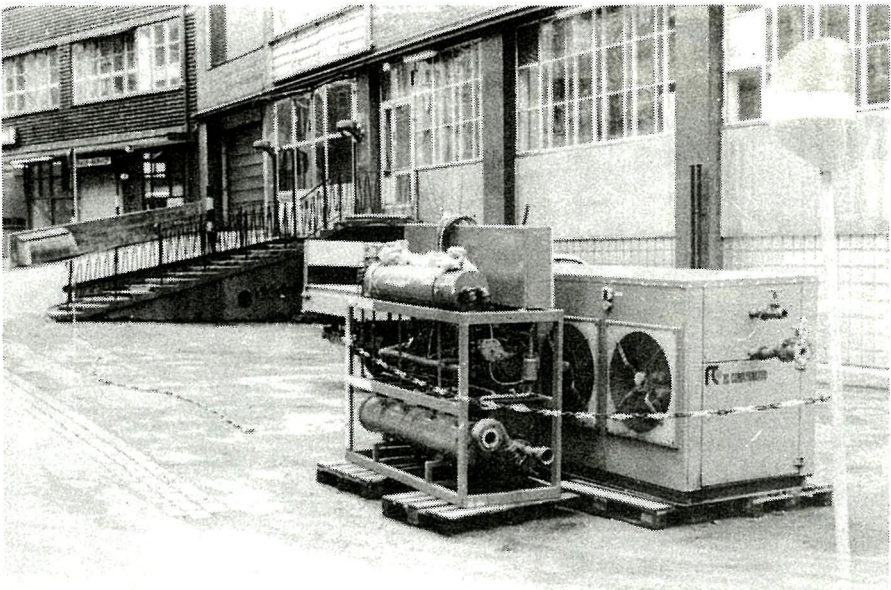


Bromsa gjorde man genom att sänka ner pallarna, dvs man fick träsplitter som bromsspår. Den första kylan, den stora RC-maskinen gick faktiskt ned riktigt fint. Sista biten var värst, för där lutade rampen minst 25 % och den var full av grus, man riskerade att ramla och slå sig redan utan last. Därför gick vi och sopade rampen, i ett moln av stendamm. Efter att vi tränat på att rulla ned RC-maskinen, ca 1,5 ton, var det inga som helst problem att rulla ned Bitzer-konstruktionen. Visserligen var den högre och kunde bara sitta på två pallar, så stabiliteten var väl inte den bästa. Det värsta verkade vara att köra ner en vanlig pall med pallkrage, lastad med ventiler. Nere på gatan parkerade vi kylarna bredvid varandra, på ett

avspärrat område.

Dagen efter ringde jag direkt till den vänliga kylfirman och frågade om kylarna var värda något som inbyte vid köp av begagnade kylmedelskylare. Jovisst, vi gjorde upp att kylfirman skulle leverera 2 st stora Asarum kylmedelskylare på KTH och i inbyte för en Asarummare fick de kylarna på gatan, Hiross-fläkten från HP, Hiross U76 från Bankgirot, en liten fläktare från Sv Finans och kylmedelskylaren vi haft på baksidan av lagret alldeles för länge. Nu var det inga problem att bli av med prylarna. På så sätt fick vi 2 st kylmedelskylare till KTH, varav en alltså är på kredit. Någon som känner för att skaka fram pengar till den, i så fall är det kr 35.000.

De två kylanläggningarna.





Nu hade vi i princip 3 kylmedelskylare, en stod ju kvar på taket hos Svenska Finans. Ja, vi hade ju tagit ut de kylar vi skulle ha ut, så nu var det bara att ringa Svenska Finans och be dem beställa ett lyft ned. De gjorde det snabbt, några dagar senare ringde Per Alm från Ostermans och vi åkte dit och tittade på om det var ett lämpligt uppdrag. Jodå, inga problem, men vi kunde inte få den levererad till KTH med helikoptern — man får inte flyga med hängande last. Vi siktade in oss på att ställa ner kylaren på parkeringen bredvid. Det visade sig vara

Olle Betzén och en av assistenterna från Ostermans kopplar lyftstroppar.

Danny Kohn kopplar på lyftvajern.

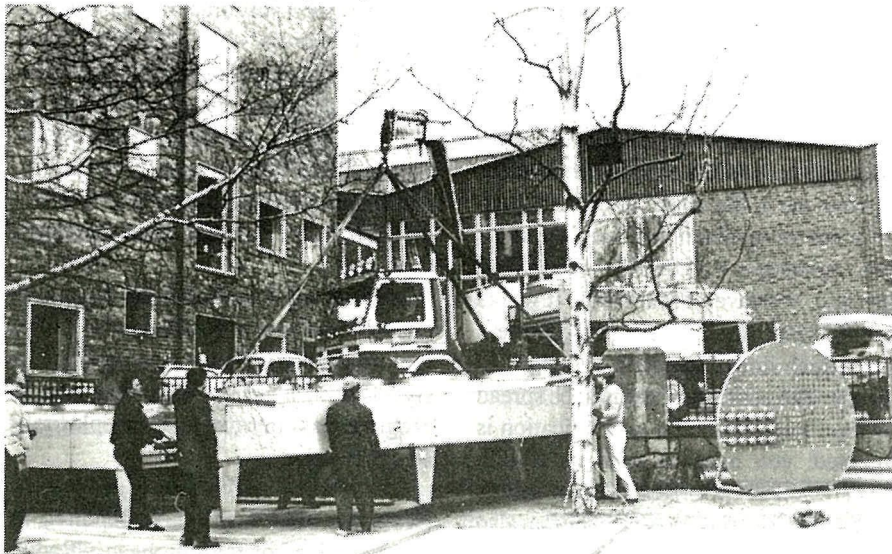


Siemens parkering och när vi sade att det gick bra att lyfta på en helg, så hade de inget att invända mot att vi spärrade av parkeringen. Jag fick låna avspärrningslinor och bockar av MFO i Älvsjö och spärrade av parkeringen en fredag kväll. Det gick bra, bara en bil stod kvar, han använde väl parkeringen som semesterparkering.

På lördagen kollade jag med piloten, jo, det skulle nog gå bra, det blåste inte alltför mycket. När vi kom dit, hade det börjat blåsa mer och hagla lite också. Det fick givetvis inte blåsa för mycket, då kan man inte helikopterlyfta och det får inte heller regna, då ser inte piloten tillräckligt bra ut ur kabinen. Men efter en liten väntan dök assistenterna

upp och vi satte igång att koppla lyftstroppar. Några fick vara på taket, några på gatan för att spärra av den när helikoptern skulle flyga över gatan, och några på parkeringen för att ta emot kylaren. Efter ett tag hördes helikopter i luften, Danny kopplade på lyftvajern och så var det bara att lyfta. Det var inga speciella problem, utom det att en av deras radioapparater dog, batteriet pajade. När han skulle sätta ned kylan blev det nedsug, det blir tydligen ofta det bakom hus och bergväggar, så han var tvungen att sätta ned kylan, det gick inte att flyga och fundera var man skulle sätta ner den. Sedan ringde vi till den kontrakterade kranbilen, som hämtade och så placerade vi ut kylarna snyggt på KTH.

Kylmedelskylarna ställs på plats på KTH.



French Virus Alert

by Thierry Delettre, Chief Sysop, Calvacom, Sun Feb 5 22:58:57 1989



Until now, all known Macintosh viruses could be easily detected by the additional resources they created. Now, it's over... There is at least one virus that creates no additional resource. This virus is called ANTI, and infects only applications (and other files, ID=1 resource). It inserts a JSR at the beginning of the resource and all the virus code at the end. It seems to be very recent, but we have already found infected Macintoshes in Paris and Marseilles, and it is probably making its way fast across all Europe. This virus is *not* detected by VirusDetective or other utilities. It installs itself even when Vaccine is on. Vaccine beeps only if the 'Always compile MPW Ints' is not checked. Virus Rx does not detect ANTI's presence in other files, but, when infected itself, changes its name to "Throw me in the trash". It doesn't seem to infect all applications, but only some (the ones with a CODE 1 resource called 'Main'). We haven't found how it works yet. It doesn't seem to change the System file, which doesn't contain a CODE resource. The contagion seems to be spread by the Finder. To see if an application is infected, you have to open its CODE ID=1 resource with ResEdit and search for the ASCII string 'ANTI'. You can also use the advanced features (resource fork search) of GOfier. We haven't yet found

the way to remove it, but only a way to deactivate it by changing the first words of the virus code to a RTS.

There is a strange story about this virus. Two years ago, Apple France's developer's support manager, Alain Andrieux, wrote a utility for his own use called 'Stamp', with which he marked the programs he gave to developers. If a confidential program was given out, he could easily know where it came from. His program added a CODE resource to the marked files, but did *not* change anything in the CODE 1 resource. In January 89, a 'new' version of this program (Stamp 1.0b5) began to spread in the French Mac community. When run, this program installs the 'ANTI' virus into the marked or checked applications and/or into the Finder. These infected applications and Finders then become contagious themselves.

It seems the virus author stole the source code of this program, changed it into a virus installer, then gave it away. Obviously, inserting a virus installer in an Apple program was done to damage Apple France's reputation...

[Note that this is the virus reported in some versions of the European "Redit" program. You have been warned. Larry]

GNU's Bulletin, Jan '89

by R Smith, P Rubin, R Chassell, L H Tower Jr., R Stallman, O Goldstein, E Suvasa

The GNU's Bulletin is the semi-annual newsletter of the Free Software Foundation, bringing you news about the GNU Project.



GNU's Bulletin

Copyright © 1989 Free Software Foundation, Inc.
675 Massachusetts Ave
Cambridge, MA 02139, USA
Telephone: +1 (617) 876-3296
Electronic mail: gnu@prep.ai.mit.edu

Permission is granted to anyone to make or distribute verbatim copies of this document as received, in any medium, provided that the copyright notice and permission notice are preserved, and that the distributor grants the recipient permission for further redistribution as permitted by this notice.

GNU's Who

Randy Smith has joined us as a full-time programmer. He is currently maintaining and extending GDB. Our summer people, **Pete TerMaat**, **Phil Nelson**, and **Mike Haertel**, have returned to school — to study or to teach. Pete worked on GDB, Phil on the GNU versions of 'cpio' and 'dbm', and Mike, who continues to work for us part time, on 'diff', 'egrep' and 'sort'.

Joe Arceneaux spent a couple of weeks with us this fall making Emacs version 19 work with X windows version 11.

Nobuyuki and **Mieko Hikichi** continue with us on loan from Software Research Associates in Tokyo. At FSF, Nobu is extending GDB with a C interpreter that he is writing. Mieko is helping user-test GNU documentation and is translating some of it into Japanese. **Diane Barlow Close**, our first full-time technical writer, is writing the documentation for all of the small Unix utilities that have been completed for us, while living in San Diego, CA.

Meanwhile, **Brian Fox** is still working for us at UC Santa Barbara. He recently completed GNU's version of 'sh', the 'Bourne Again Shell', that incorporates features found in the Korn and C shells. **Jay Fenlason** is writing a spreadsheet program for the project and maintaining the GNU assembler, 'tar', and 'sed'.

Opus Goldstein is our jack-of-all-trades

office staff. If you call our office, she is the one who answers. She fills the orders, and handles the day-to-day operations of the Foundation. **Robert Chassell** is our Treasurer. Besides dealing with corporate issues not related to programming, he is working on an elementary introduction to programming in Emacs Lisp.

Richard Stallman continues to do countless tasks, including refining the C compiler, GNU Emacs, etc. and their documentation. **Paul Rubin** is writing a graphic editing extension for GNU Emacs. Finally, **Len Tower** continues as our electronic JOAT (jack-of-all-trades), handling mailing lists, information requests, system mothering et al.

What Is the Free Software Foundation?

The Free Software Foundation is dedicated to eliminating restrictions on copying, redistribution, understanding and modification of computer programs. We do this by promoting the development and use of free software in all areas of computer use. Specifically, we are putting together a complete integrated software system called "GNU" (GNU's Not Unix) that will be upward compatible with Unix. Some large parts of this system are already working and we are distributing them now.

The word "free" in our name refers to two specific freedoms: first, the freedom to copy a program and give it away to your friends and co-workers; second, the free-

dom to change a program as you wish, by having full access to source code. Furthermore, you can study the source and learn how such programs are written. You may then be able to port it, improve it, and share your changes with others.

Other organizations distribute whatever free software happens to be available. By contrast, FSF concentrates on development of new free software, building toward a GNU system complete enough to eliminate the need to purchase a proprietary system.

Besides developing GNU, the Foundation has secondary functions: producing tapes and printed manuals of GNU software, carrying out distribution, and accepting gifts to support GNU development. We are tax exempt; you can deduct donations to us on your tax returns. Our development effort is funded partly from donations and partly from distribution fees. Note that the distribution fees purchase just the service of distribution: you never have to pay anyone license fees to use GNU software, and you always have the freedom to make your copy from a friend's computer at no charge (provided your friend is willing).

The Foundation also maintains a Service Directory: a list of people who offer service for pay to individual users of GNU programs and systems. Service can mean answering questions for new users, customizing programs, porting to new systems, or anything else. Contact us if you want to be listed.

After we create our programs, we continually update and improve them. We release between 2 and 20 updates a year, for various programs. Doing this while developing new programs takes a lot of work, so any donations of pertinent source code and documentation, machines, labor or money are always appreciated.

What Is Copyleft?

In the article "What Is the Free Software Foundation," we state that "you never have to pay anyone license fees to use GNU software, and you always have the freedom to make your copy from a friend's computer at no charge." What exactly do we mean by this, and how do we make sure that it stays true?

The simplest way to make a program free is to put it in the public domain. Then people who get it from sharers can share it with others. But bad citizens can also do what they like to do: sell binary-only versions under typical don't-share-with-your-neighbor licenses. They would thus enjoy the benefits of the freeness of the original program **while withholding these benefits from the users. It could easily come about that most users get the program this way, and our goal of making the program free for all users would have been undermined.**

To prevent this from happening, we don't normally place GNU programs in the public domain. Instead, we protect them by what we call *copyleft*s. A copyleft is a

legal instrument that makes everybody free to copy a program as long as the person getting the copy gets with it the freedom to distribute further copies, and the freedom to modify their copy (which means that they must get access to the source code). Typical software companies use copyrights to take away these freedoms; now we software sharers use copyleft to preserve these freedoms.

The copyleft used by the GNU project is made from a combination of a copyright notice and the *GNU General Public License*. The copyright notice is the usual kind. The General Public License is a copying license which basically says that you have the freedoms we want you to have and that you can't take these freedoms away from anyone else. (The actual document consists of several pages of rather complicated legalbol that our lawyer said we needed.) A copy of the complete license is included in all GNU source code distributions and many manuals, and we will send you a printed copy on request.

GNU in Japan

by Mieko Hikichi

The GNU Project was described in a seminar at the Unix Fair in Tokyo last December. This seminar was the first official introduction of GNU in Japan.

I had translated the GNU's Bulletin into Japanese and 500 copies were distributed.

Mr. Yoshitaka Tokugawa talked about what is GNU, how to get GNU software and about the GNU license. This was followed by a talk about the Japanese version of GNU Emacs by Mr. Handa Ken'ichi.

The General Public License as a Subroutine

We are about to make a sweeping, revolutionary change in the General Public License. The terms for copying will be essentially unchanged, but the architecture of the legalbol framework used to embody them will make a quantum leap.

In the past, each copylefted program had to have its own copy of the General Public License contained in it. Often it was necessary to modify the license to mention the name of the program it applied to. Other people who wanted to copyleft programs had to modify the text even more, to replace our name with theirs.

To make it easier to copyleft programs, we ~~have been improving on the legalbol architecture of the~~ General Public License ~~to produce a new~~ version that serves as a ~~general-purpose~~ subroutine: it can apply to any program without modification, no matter who is publishing it. All that's needed is a brief notice in the program itself, to say that the General Public License applies. Directions on doing this accompany the General Public License, so you can easily copyleft your programs.

We've also taken the opportunity to make it explicit that any subsequent changes in future versions the General Public License cannot take away the rights you were previously given, if you have a program that you received under an earlier version.

The new version will appear soon; new GNU programs and new versions of existing GNU software will refer to it to specify their copying conditions.

GNUsworthy Flashes

* Donation from Hewlett-Packard

As part of their program to make their machines more attractive to academic users, Hewlett-Packard is giving us \$100,000. This money was given to us to make sure that we have funds to rent office space for several years. Of course, if someone were to give us office space, we could use the money to hire more people.

In addition, the company is supporting Michael Tiemann's work at Stanford to port G++ to the Spectrum.

* New Machines

Since the last GNUs Bull in June 1988, we have received three new machines on loan and three as gifts. Thinking Machine, Inc. has loaned us a Sun 4/110, and Intel has loaned us an Intel 386. K. Richard Magill has given us an AT&T Unix PC and Hewlett-Packard is giving us three Spec-

trums in addition to the funds mentioned above. This makes for a total of 13 machines given or loaned to FSF.

* Some parts of BSD are becoming free

After years of urging from us and others, the people who maintain Berkeley Unix have decided to release various parts of it (those which don't contain AT&T code) separately as free software. This includes substantial programs which we hope to use in GNU, such as TCP/IP support and possibly the C-shell.

The files freed in the 4.3 BSD-Tahoe distribution are now on the beta test tape.

Also, the next release of Berkeley Unix may contain the Make, AWK, and 'sh' from the GNU Project instead of those from Unix. The reason is that the people at Berkeley would like to have improvements in these programs like those in system V.3, but they find the new restrictions on V.3 licenses unpalatable. Both we and they hope they never get a V.3 license. We may help them avoid it by providing alternative software.

GNU Make already supports the system V features, as does GAWK. Both programs are in beta test.

* The Bison manual is here

Dick Karpinski of UCSF offered a prize of \$1000 for a usable draft of a manual for Bison (our species of Yacc). Chuck Donnelly responded with a draft which we

have turned into a final version. Texinfo source comes on both the release and beta test tapes. Hardcopy may be ordered directly from the Foundation.

*** Ghostscript status**

Ghostscript, the free Postscript for GNU, is being distributed. It is now in beta test.

*** NeXT, Inc. is using GNU software on their machines**

NeXT has decided to use GNU Emacs, GCC, GAS and GDB as part of their standard operating system; they are the first computer manufacturer to do so. Other manufacturers have expressed an interest; we are awaiting developments.

*** Distribution of 80386 Floppies on hold**

We are holding investigating distribution of floppies for 80386 machines until we have a working library. Since system V on the 80386 usually comes with no library as well as no C compiler, no assembler and no linker, we don't think this distribution medium will be worth while until we can supply all of them.

This is being worked on, but it will not be ready very soon. We will announce on the `info-gnu` mailing list and `gnu.announce` newsgroup when we have more information.

*** texi2roff**

Beverly Erlebacher of Toronto, Canada wrote 'texi2roff'. This program translates GNU Texinfo files into a format that can be printed by the Unix '[nt]roff' programs utilizing the 'mm', 'ms' or 'me' macro packages. Though we won't be maintaining this program, we include it on our tapes to allow people without T_EX to print out our documentation.

GNU Wish List

Wishes for this issue are for:

✓ Volunteers to help write utilities and documentation. One important programming project is writing a Fortran compiler front end for GCC; documentation particularly needed includes manuals for Ghostscript and FLEX.

✓ FSF is looking to hire additional full-time people to work on Project GNU both as programmers and as technical writers. We are looking for a few good entities (male or female, carbon or silicon based) to work for an activist wage. You must either be in Cambridge, Mass or be able to maintain good electronic communication with us. Contact

`rms@wheaties.ai.mit.edu`
or send mail to Richard Stallman c/o the Free Software Foundation if you are interested.

✓ Office space (lent or donated) in or around Kendall Square.

✓ Professors who might be interested in sponsoring research assistantships with full or partial FSF support to do GNU development, or hosting FSF employees.

✓ Money, as always. Please remember, donations are tax-deductible. With the latest donations, we have been able to expand our staff.

One way to give us a small amount of money is to order a distribution tape or two. This may not count as a donation for tax purposes, but for many people it can qualify as a business expense.

✓ Someone skilled in compiler maintenance who could take over GCC maintenance for RMS. This would probably be a full-time job.

✓ Interesting, fun GNU graphics. Ghostscript source or otherwise. See *GNU "ClipArt" Contest* later in this issue for more details.

✓ Copies of newspaper and journal articles mentioning the GNU Project. Send these to the addresses on the front cover.

My Thoughts on the GNU License

by Doug Lea

[This article is reproduced from a posting to the info-g++ and info-gcc mailing lists and does not necessarily represent the opinions of the Free Software Foundation. – Editor]

I have not participated publicly in the recent discussions about the legal ramifications of the GNU License Agreement until now because (1) I am not a lawyer and (2) I find myself in agreement with Stallman's decision to proceed very carefully in deciding whether and how to modify the Agreement to accommodate people who would like to sell works that may or may not be considered as 'derived' from GNU software, depending on what 'derived' is defined to mean.

However, the recent proposal by Gilmore and others appears to demand a personal response from me (not RMS or FSF) as the author of most of GNU 'libg++'. I would like to briefly outline why I support FSF goals, and specifically, the Agreement, in a way that bears only indirectly on legal principles.

I am, primarily, a teacher in a liberal arts college. As such, I stand for the 'free' dissemination of ideas. Historically, (please forgive any botching of historical facts to suit my needs, but that's what history is for!) the main tool by which intellectual property has been allowed to be widely disseminated (read 'taught') while at the same time both crediting originators, and protecting the works from corruption, mis-attribution, and so on, has been the notion of Copyright. For these reasons, the introduction of copyright laws is widely considered to have been an important step in accelerating intellectual and scientific progress.

Sadly, in the science of computing, this

solution has not stood up well. While, in many disciplines, the price of a copyrighted work to be used for study is well within the reach of those who could best benefit from it (e.g., a copy of "War and Peace" might be \$5, or even \$50, but not \$50,000), the economics of computing have, for the most part, priced copyrighted software out of the reach of students (and most others). Most readers would agree that the study of high-quality existing programs is among the better methods for learning about the art of programming. These days, one cannot legally show, discuss, and teach from, say, Unix or Lotus source code.

I believe that Stallman's notion that the economics of copyright can be separated from its role in the protection and propagation of intellectual property is as good a solution to this dilemma as we are likely to get. There are many of us, especially those of us in academe, who are actually very pleased to devote some time and effort to writing software without any direct monetary compensation. For all sorts of reasons. (For example, in my case, with 'libg++', as a means to further investigate the pragmatics of object-oriented programming and so on. Or maybe it's just incorrigible hacking. Whatever.)

Now I, and many others, I suspect, are not terribly worried about maintaining proper authorship credit, etc., of such work. The reason that the GNU License Agreement is attractive is mainly that it keeps accessible the work that I intended to be acces-

sible, but also generally offers all other benefits that Copyright engenders, but that the mere act of placing work in the 'public domain' would not.

It is an unfortunate fact that the GNU Agreement currently stands in the way of such work being used in honest ways by honest programmers who do have to worry about the economic ramifications. I personally hope that exactly the right accommodations are made to allow fair and sensible use while maintaining the ideals that make the GNU solution work. There are many sticky legalistic points involved in doing so. I do hope that Stallman is able to find such a solution soon enough to make alternative approaches less attractive, but not so hastily as to compromise the goals of FSF (which I am sure he will not do).

GNU Documentation

Software distributed as part of the GNU system always comes with sources for both on-line documentation and printable manuals. On-line documentation is provided because printed documentation is often misplaced or is being read by someone else when you want it; and because, in many situations, it is easier to find the piece of information that you need in an interactive on-line help system. At the same time, printed documentation is provided because it is often easier to read or preferred.

When GNU documentation is printed, it

is produced as a typeset book with chapters, indices, cross references and the like. The on-line documentation is a menu-driven system which also uses indices and cross references.

However, instead of writing two different documents, one for the on-line documentation and the other for the printed manual, GNU documentation uses a single Texinfo source file for both purposes. This saves the effort of writing two different documents and means that when the system is revised, only one source file has to be revised.

Since the single Texinfo source file is used for a dual task—to create both the on-line documentation and the printed manual—it must be written in a special format so that the chapters and sections of the printed manual will correspond to the nodes of the on-line documentation and the indices and cross references will correspond to various menus.

To make the printed manual, the Texinfo source file is processed through the `TEX` typesetting program. To make the on-line documentation, using GNU Emacs, the Texinfo source file is processed with the `M-x texinfo-format+buffer` command; the resulting Info file is installed in the `info` directory which you reach by typing `C-h i`.

(Non-GNU software distributed by the Free Software Foundation does not always have Texinfo documentation, although we encourage everyone to docu-

ment with it.)

All of the following manuals, which we are currently distributing on our tapes, are also available in hardcopy from the Foundation; see the order form on the inside back cover.

❖ Emacs Manual

The *Emacs Manual* describes how to use GNU Emacs. The manual not only explains Emacs' editing features, but it also explains advanced features, such as outline mode and regular expression search. The manual tells how to use the special modes for programming in languages such as C and Lisp, how to use the tags utility, and how to compile and correct code. It also describes how to make your own keybindings and make other elementary customizations.

❖ Texinfo Manual

The *Texinfo Manual* describes how to write documents in Texinfo source code. It describes the markup language used to create both an Info file and a printed document from the same source file: how to make tables and lists, how to make chapters and nodes, how to make indices and cross references. The manual also describes how to use Texinfo Mode in GNU Emacs and how to catch mistakes.

❖ Termcap Manual

The *Termcap Manual* is often described as "Twice as much as you ever wanted to

know about Termcap". The manual describes the format of the Termcap data base, the definitions of terminal capabilities and how to find and interrogate a terminal description. This manual is primarily for programmers.

❖ Bison Manual

The *Bison Manual* describes how to write a grammar description that Bison can convert into a C program that can parse that grammar. It describes the concepts and then provides a series of increasingly complex examples before describing what goes on in considerable detail.

❖ GDB Manual

The *GDB Manual* explains how to use the GNU DeBugger. It describes how to run your program under control of the debugger, how to examine and alter data within the program, and how to modify the flow of control within the program. It also explains how to use GDB through GNU Emacs, with auto-display of source lines.

❖ GAWK Manual

The *GAWK Manual* describes how to use the GNU implementation of AWK. It is written for someone who has never used AWK before, and describes all the features of this powerful string manipulating language.

❖ Make Manual

The *Make Manual* describes the GNU

Make utility, a program used to recompile parts of other programs when and as needed. The manual tells how to write a Makefile, which describes how to recompile the parts of your program and how they depend on each other.

GNU Project Status Report

▼ GNU Emacs

GNU Emacs 18 is now being distributed. It is in wide use on several kinds of BSD 4.2 and 4.3 systems and on system V, VMS and Apollo Domain. Version 18.52, the current version, adds support for the 80386, the Sun 4, the Convex, the IRIS 4D and the HP 9000 series 800; also support for system V.3.

Berkeley is distributing GNU Emacs with the 4.3 distribution, DEC is distributing it with Unix systems on Vaxes, and NeXT is shipping it.

Emacs 18 maintenance is about to be finished, and version 19 is being worked on. New features so far include multiple X windows, scroll bars, per-buffer redefinition of mouse commands, support for European character sets, source-level debugging for Emacs Lisp, and floating point numbers. Planned possible new features include:

- ✓ Associating property lists with portions of the text in a buffer.
- ✓ Using the properties to control which text is visible.

- ✓ Different visibility conditions for many windows showing one buffer.
- ✓ Incremental syntax analysis for various programming languages.
- ✓ An object-oriented graphics-drawing extension.
- ✓ Hooks to be run if point moves outside a certain range.

Once I get time to work on it, it will take several months of work to do this, so don't ask when Emacs 19 will be available for beta test. We will announce it.

▼ Improved GAWK available

David Trueman has written the new System V extensions to GAWK and the results are now available on the beta tape. Meanwhile, a manual for GAWK is on its way.

▼ Shells

Brian Fox has now completed GNU's version of 'sh', called the 'Bourne Again Shell'. In addition to Korn shell features, it now has job control and both Emacs-style and 'csh'-style command history.

There is a good chance that the C Shell from BSD will be declared free software by Berkeley, so we won't need to write that.

▼ Kernel

We hope to use the MACH message-passing kernel being developed at CMU. The current version of MACH is not free,

and cannot be, because it contains the file system code from BSD. However, the MACH developers say that all this will be replaced with free code, or at least moved into user processes, and MACH will be free then. This version of MACH is supposed to be released in a few months (as of December 1988).

If MACH does not become available, then we will probably develop the GNU kernel starting with either MIT's TRIX kernel or Berkeley's Sprite system. TRIX is a remote procedure call kernel that runs and supports basic Unix compatibility at about the level of version 7. So it needs a lot of additional features. Sprite is mostly at the architectural level of BSD Unix, but with a fancy distributed file system and process migration.

One thing we are considering is adapting the file system from Berkeley's Sprite kernel for use in MACH. This file system was designed from the beginning to work in a distributed manner. The file system is the largest part of MACH that needs replacement, now that the Berkeley TCP/IP code, also used in MACH, has been declared free.

▼ GNU DeBugger

The GNU source-level C debugger, GDB, is now being distributed along with Emacs version 18. Currently it runs under BSD 4.2 and 4.3 on Vaxes and Suns. It can also run stand-alone so we can use it to debug the kernel. An over-the-ethernet debugging mode may be added. Work is

being done on debugging of multiple process parallel programs.

GDB support for the 32000 is in the current version. GDB can now read COFF executables, and it is just starting to work on system V.

GDB 3.0, recently released for testing, can read symbol tables incrementally; this results in much faster startup and much less memory use. It also has C++ support and can run on the 80386 and the Sun 4, as well as on some Gould machines.

Future versions of GDB may support watchpoints, and command editing with identifier completion.

▼ C Compiler

The GNU C Compiler GCC is now nearly reliable. It supports the May 1988 draft of ANSI C. People are still reporting bugs, but they also say they think there are fewer bugs than in commercial compilers. Some of these bugs were uncovered by John Gilmore, who has been compiling all of BSD Unix with GNU C. A review comparing GCC with two commercial C compilers appeared in the March, 1988 issue of Unix Review magazine.

The compiler performs automatic register allocation, common subexpression elimination, invariant code motion from loops, induction variable optimizations, constant propagation and copy propagation,

delaying popping of function call arguments, tail recursion elimination, plus many local optimizations that are automatically deduced from the machine description.

CPUs supported include the 680x0, Vax, 32x32, 80386, Sparc (Sun 4), SPUR and Alliant. Ports for the Convex, the IBM 370, the IBM PC/RT, the 3b2, some sort of Gould machine, the HP Spectrum, the AMD 29000, the MIPS RISC chips, and the Motorola 88000 are on their way.

GCC makes shorter and faster 68020 code than the new Sun compiler with -O. The new Sun compiler can't beat GCC despite taking 3 times as long. As of version 1.31, GCC also wins on the Sun 4. GCC makes shorter Vax code than the Tartan C compiler with -O4, but we haven't been able to compare the running speed of that code. A National 32000 port done by Michael Tiemann on a Sequent 32000 system is said to be more reliable than Sequent's compiler and yields a 40% speedup for several programs including a Prolog interpreter.

In the future, if we have time, we would like to implement some more language extensions, plus facilities for precompilation of header files to save time when the headers are large and the source files are small. We might also do some other language front ends, but there seems to be enough interest among remote GNUers in doing this that we might as well leave it to them.

Enough internal documentation is included for people interested in retargeting the compiler to other CPUs to do so.

▼ GNU C for VMS

Fed up with the deficiencies of the VMS C compiler, David Kashtan from SRI decided to spend a couple of weeks and make GNU C run on VMS. After making considerable changes to satisfy the VMS C compiler, he got it running and was able to take most of the changes out. The VMS support code is now part of the regular compiler distribution.

The ordinary VMS C compiler (even if you have it) has bugs and therefore cannot compile GNU C: you need an executable of GNU C. We offer mag tapes containing VMS binaries of GNU C (as well as sources) so you can get started.

Other GNU programs currently working on VMS include GNU Emacs and Bison.

Please don't ask us to devote effort to additional VMS support, because it is peripheral to the GNU project. We merge in and support VMS ports that users do, because it is hard to refuse to pass along work that other people have done. But even when the changes are clean, this drains considerable effort from our real goal, which is to produce a complete integrated system. (When they aren't clean, we summon up the courage to ignore them.) Merging VMS GNU Emacs and reorganizing the changes to ease future maintenance consumed several weeks

even though the "real work" was done by others. We hope we have learned not to let this happen again.

▼ COFF support

It is now possible to run the entire suite of GNU software tools on system V, replacing COFF entirely.

First you install the GNU compiler, assembler, linker and other utilities. Then you use the program 'Robotussin' – COFF medicine for your computer – to convert the system libraries from COFF format to GNU (i.e. BSD) format.

When you compile programs, you will get ordinary GNU/BSD object files. Linking these with the GNU linker will produce GNU/BSD executables with a small amount of COFF header material to encapsulate them so that the kernel can run them. The other GNU utilities such as 'size', 'nm' and 'strip' know how to operate on these encapsulated files.

We can't afford to take the time to implement or even maintain actual COFF support in these utilities. Completing the GNU system is more important to us.

▼ Compiler-related programs

▼ C'++'

Michael Tiemann of Stanford University (once of MCC) has written a C++ compiler as an extension of GNU C. This is the first compiler that compiles C++ directly

instead of preprocessing it into C.

A review comparing G++ with two commercial C++ compilers appeared in the November, 1988 issue of Unix Review magazine. The high marks convinced us it was time to start beta testing, so C++ is now included on the beta tape.

We are now working on integrating it with the C compiler; the design issues have been solved and a merged version will probably be released by the end of February.

GDB version 3.0 includes support for debugging C++ code, which will make the program GDB+ obsolete.

▼ Assembler

We have a partially-portable one pass assembler, GAS, that is almost twice as fast as the Unix assembler. It now works for Vaxes, 680x0, 32x32 and 80386. Porting it to other machines should not be hard.

▼ C++ library

Doug Lea is writing 'libg++', a library including utility classes support for C++.

▼ Object file utilities

The GNU replacements for 'ld', 'nm', 'size', 'gprof', 'strip', 'ranlib', etc., were recently released for testing as part of the GCC test distribution.

The GNU linker 'ld' runs significantly faster than the BSD version. Our 'ld' is the only one that will give you source-line numbered error messages for multiply-defined symbols and undefined references.

▼ 'gprof' replacement

Foundation staffer Jay Fenlason has completed a profiler to go with GNU C, compatible with 'gprof' from Berkeley Unix. It is now in the binary file utilities release.

▼ GNU Make extended

We have been distributing the GNU Make for several months. An extended version, which includes more text-processing capabilities and static rules, will be released soon.

Here is how a GNU Makefile can say that the file 'foo' is linked from the object files of all C source files in the current directory:

```
objects:=$(subst .c,.o,$(wildcard *.c))
foo: $(objects)
    $(CC) -o foo $(objects) $(LDFLAGS)
```

GNU Make also supports parallelism.

▼ C library

Roland McGrath, who contributed a great deal to GNU Make, has a nearly complete set of ANSI C library functions. We hope they will be ready some time this spring. These join the GNU 'malloc', 'regexp'

and Termcap libraries that have existed for some time. A newer, better 'malloc' may be written soon.

Meanwhile, Steve Moshier has contributed a full series of mathematical library functions.

▼ Preliminary Ghostscript available

We have just started to distribute Ghostscript, the free GNU software that provides nearly all the facilities of a Postscript interpreter. It supports X version 11. Peter Deutsch is still doing some work on it.

Right now, Ghostscript will accept commands in Postscript and execute them by drawing on an X window. It needs enhancement:

- ✓ to serve as a previewer for multi-page files.
- ✓ to serve other X clients by drawing on their windows.
- ✓ to improve both the performance and the visual quality of the output.
- ✓ more fonts.

Ghostscript also includes a C-callable graphics library (for client programs that don't want to deal with the Postscript language), and also supports IBM PCs and compatibles with EGA graphics (but please don't ask the FSF staff any questions about this; we don't use PCs and don't have time to learn anything about them).

▼ Finger

We will have a daemon-based Finger program soon. The daemon polls a selected set of hosts and thus is able to tell you where each person is logged in.

▼ Send

We are now trying out a Send program for sending immediate messages to other users on the same machine or other machines.

▼ GNU mailer

Landon Noll and Ronald Karr of Amdahl are writing a mail queuing and delivery system, called 'smail'. This project will be a supported part of the Amdahl UTS system—and it will be available on exactly the same terms as GNU Emacs!

We may use this mailer for the GNU system, or another mailer, 'zmailer', that Rayan Zachariasen is writing, or a combination of the two.

▼ Window system

We plan to use the X window system written at MIT. This system is already available free.

▼ Other utilities

The GNU 'ls', AWK, Make and 'ld' are in regular use. Our free replacement for Yacc is called Bison. We also have 'cron'. We now have the world's fastest 'grep'.

'egrep' and the world's fastest 'diff'. A new fast 'sort' has just been finished.

A "fast lex" called FLEX recently became available; we are now distributing it.

▼ Documentation system

We now have a truly compatible pair of programs which can convert a file of Texinfo format documentation into either a printed manual or an Info file.

See *GNU Documentation* earlier in this issue for more information.

▼ Possible target machines

The GNU Operating System will require a CPU that uses 32-bit addresses and integers and addresses to the 8-bit byte. 1 megabyte of core should be enough, though 2 meg would probably make a noticeable improvement in performance. Running some of the system in 1/2 meg may be possible, but certainly not GNU Emacs. GNU Emacs requires more than a megabyte of addressable memory. We expect that virtual memory will be required.

A hard disk will be essential; at least 20 meg will be needed to hold a minimal system. Plus more space for the user's files, of course. We recommend at least 80 meg for a personal GNU system.

This is not to say that it will be impossible to adapt some or all of GNU for other kinds of machines; but it may be difficult,

and We don't consider it part of our job to try to reduce that difficulty.

We have nothing to say about any specific models of microcomputer, as we do not follow hardware products.

▼ Porting

It is too early to inquire about porting GNU (except GNU Emacs and GNU C). First, we have to finish it.

GNU "Clip Art" Contest

We are looking for freely redistributable art work and graphics to enhance our publications. The art should be about the GNU Project or the free software movement. We offer a GNU Manual of the artist choice, for each piece of art work which we publish. Ghostscript source or more traditional media welcome. Send submissions to one of the addresses on the front cover. Our thanks to the Icon Project at the University of Arizona for the idea of running this contest.

GNU Software Available Now

We now offer three Unix software source distribution tapes, plus VMS tapes for GNU Emacs and GNU C that include sources and VMS executables. The first Unix tape (called the "Release" or "Emacs" tape) contains GNU Emacs as well as various other well-tested programs that we consider reliable. The se-

cond ("Beta test" or "Compiler") tape contains the GNU C compiler and related utilities, and other new programs that are less thoroughly tested. The third ("X11") tape contains the X11 distribution from the MIT X consortium. Until recently, this software had been provided on the Beta test tape, but the third release of the X11 distribution is too big to fit on this tape. See the order form for details about media, etc.

Contents of Release tape

* GNU Emacs

In 1975, Richard Stallman developed the first Emacs: the extensible, customizable real-time display editor. GNU Emacs is his second implementation of Emacs. It's the first Emacs available on Unix systems that offers true Lisp, smoothly integrated into the editor, for writing extensions. It also provides a special interface to MIT's free X window system, versions 10 and 11, which makes redisplay very fast. The current version is GNU Emacs 18.52; version 18.53 should be released by the time this bulletin sees press.

GNU Emacs has been in widespread use since 1985 and often displaces proprietary implementations of Emacs because of its greater reliability as well as its good features and easier extensibility.

GNU Emacs (as of version 18.52) has run on many kinds of Unix systems: those made by Alliant (system releases 1 to 4),

Altos 3068, Amdahl (UTS), Apollo, AT&T (3b machines and 7300 pc), CCI 5/32 and 6/32, Celerity, Convex, Digital (Vax, not PDP-11; BSD, and SysV), Dual, Elxsi 6400, Encore (DPC and APC), Gould, HP (9000 series 200, 300 or 800 (Spectrum) but not series 500), HLH Orion 1/05, IBM (RT/PC running 4.2 and AIX), Integrated Solutions (Optimum V with 68020 and VMEbus), Intel 80386 (BSD, SysV, and Xenix), Iris (2500, 2500 Turbo and 4D), LMI (Nu), Masscomp, Megatest, MIPS, NCR (Tower 32), Nixdorf Targon 31, Plexus, Pyramid, Sequent Balance, SONY News, Stride (system release 2), Sun (any kind), Tahoe, Tektronix (NS16000 system & 4300), Texas Instruments (Nu), VAX (BSD, SysV, VMS) and Whitechapel (MG1).

GNU Emacs use is described by the GNU Emacs Manual, which comes as a Texinfo file with the software. You can read it on-line in Info form or print out your own copy. Typeset manuals are also available from the Free Software Foundation.

* Texinfo

Texinfo is the documentation system used for all GNU manuals. Documents are written in a simple formatting language that can produce either printed manuals using a special set of $T_E X$ macros, or on-line structured documentation that can be read using the Info browser. Included with Texinfo is a newly expanded tutorial manual on how to write documents using Texinfo.

* GDB

GDB (the GNU Debugger) is the source-level C debugger written for the GNU project in 1986. It offers many features not usually found in debuggers on Unix, such as a history that records all values examined within the debugger for concise later reference, multi-line user-defined commands, and a strong self-documentation capability. It currently runs on Vaxes under 4.2 and 4.3bsd, on Sun 3 under systems version 2 and 3 and 4, on the SPARC (Sun 4) under systems version 3.2 and 4.0, HP9K320, ISI, Merlin, Sony NEWS, Gould NPL and PN, i386, and on some 32000 systems.

On-line help and a users' manual for GDB comes with the software; the printed version of the manual is also available from the Foundation.

* Bison

Bison is an upward-compatible replacement parser generator for Yacc, with additional features. It has been in use for several years. Bison is used for compiling GNU C, so it is included on the GNU CC beta tape as well. A users' manual for Bison comes with the software; the printed version is also available from the Foundation.

* X Window System

Version 10 of X Windows is distributed on the GNU Emacs tape; version 11 (which is totally incompatible) is distributed

on the X11 tape. Emacs version 18.52 supports both versions 10 and 11.

* MIT Scheme

Scheme is a simplified, lexically scoped dialect of Lisp, designed at MIT and other universities for two purposes: teaching students of programming, and researching new parallel programming constructs and compilation techniques. MIT Scheme is written in C and runs on many kinds of Unix systems.

It now conforms to the *Revised Report On The Algorithmic Language Scheme* (MIT AI Lab Memo 848a), for which T_EX source is included in the distribution.

* T

A variant of Scheme developed at Yale University, T is intended for production use in program development. T contains a native-code optimizing compiler that produces code that runs at speeds comparable to the running speeds of programs written in conventional languages. It runs on BSD Vaxes and a few types of 68020 systems. T is written in itself and cannot be bootstrapped without a binary (included), but it is great if you can use it. Some documentation files are included in the distribution.

* GNU Chess and Hack

GNU Chess is a chess program written in C by John Stanback and Stuart Cracraft. It includes an extensive opening book and

was recently rated by USCF Senior Master IM Larry Kaufman at around USCF 1950 (close to expert level) when run on a Sun 3 workstation. On a Sun 4, it should play at nearly master level.

Hack is a display oriented adventure game similar to Rogue.

Contents of Beta Test Tape

The programs on this tape are all recent releases and can be considered to be at various stages of user testing. As always, we solicit your comments and bug reports. This tape is also known as the Compiler tape.

* GNU CC

The GNU C compiler is a fairly portable optimizing compiler. It generates good code for the 32000, 680x0, 80386, Alliant, SPARC, SPUR, and Vax CPUs. Machines using these CPUs include the Encore NS32000, Genix NS32000, Sequent NS32000, AT&T 3b1, HP-UX 68000/68020, ISI 68000/68020, Sony NEWS, AllSun's, Intel 386, Sequent Intel 386, Alliant FX/8, and Vax machines. It supports full ANSI C as of the latest draft standard. Included with the compiler are the GNU assembler GAS, Make, Bison (also on the Emacs release tape), and the perfect-hash hash-table generating utility, plus the object file utilities 'ld', 'nm', 'size', 'strip', 'ar', 'ranlib' and 'gprof' and the Texinfo source of *The GCC Manual* (for people interested in extending or

retargeting the compiler).

* GAWK and FLEX

GAWK is GNU's version of the Unix AWK utility. FLEX is a mostly-compatible replacement for the Unix 'lex' scanner generator written by Vern Paxson of the Lawrence Berkeley Laboratory. FLEX generates far more efficient scanners than 'lex' does.

* 'tar'

GNU tar includes multivolume support, automatic compression and decompression of archives, remote archives, and special features to allow 'tar' to be used for incremental and full backups of file systems.

* Freed files from the U.C. Berkeley 4.3-tahoe release

These files have been declared by Berkeley to be free of AT&T code, and may be freely redistributed. They include complete sources for some utility programs and games, as well as library routines and partial sources for many others.

* 'diff' and 'grep'

These programs are GNU's versions of the Unix programs of the same name. They are much faster than their Unix counterparts.

* Ghostscript, 'gnuplot' and Contents of X11 Tape. 'texi2roff'

Ghostscript is GNU's graphics language. It is almost fully compatible with the postscript language. It supports X version 11.

'Gnuplot' is an interactive program for plotting mathematical expressions and data. Oddly enough, the program was neither done for nor named for the GNU Project – the name is a coincidence. However, we are now distributing it.

'texi2roff' translates GNU Texinfo files into a format that can be printed by the Unix [nt]roff programs utilizing the mm, ms or me macro packages. It is included so that people who don't have a copy of T_EX can print out GNU documentation.

* 'g++', 'libg++' and 'oops'

G++ is a front-end for GCC, that compiles C++, an object-oriented language that is upward compatible with C. Since G++ depends on GCC, it can only be used with the corresponding version of GCC.

'libg++' (the GNU C++ library) is a collection of C++ classes and support tools for use with G++.

OOPS (Object-Oriented Program Support) class library is a portable collection of classes similar to those in Smalltalk-80 that has been developed by Keith Gorlen of NIH, using the C++ programming.

X is a portable, network transparent window system for bitmap displays written at MIT and DEC. It runs Sun, DEC VAXstation, and various other current bitmap displays. X supports overlapping windows and fully recursive subwindows, and provides hooks for several different styles of user interface. Applications include a terminal emulator, bitmap editor, several window managers, clock, window dump and undump programs, and several typesetting previewers.

The X11 tape contains Version 11, Release 3 of the MIT/DEC X window system. X11 is more powerful than, but incompatible with, the no-longer-supported version 10. MIT no longer labels Version 11 'beta test' but is still releasing frequent patches and updates.

VMS Emacs and Compiler Tapes

We offer a VMS backup tape of the GNU Emacs editor, and a separate tape containing the beta-test GNU C compiler. The VMS compiler tape also contains Bison (needed to compile GCC), GAS (needed to assemble GCC's output) and some library and include files. Both VMS tapes include executables that you can bootstrap from.

How To Get GNU Software

All the software and publications from the

Free Software Foundation are distributed with permission to copy and redistribute. The easiest way to get a copy of GNU software is from someone else who has it. Just copy it from them.

If you have access to the Internet, you can get the latest software from the host `prep.ai.mit.edu`. For more information, read the file `/u2/emacs/GETTING.GNU.SOFTWARE` on that host.

If you cannot get the software from a friend or over the net, or if you would feel more confident getting copies straight from us, or if you would like to contribute some funds to our efforts, the Free Software Foundation distributes tapes for a copying and distribution fee. See the order form on the inside back cover.

If you do not have net access, and your computers cannot use the media we distribute on, you must get our software from third party groups — people and organizations that do not work with us, but have our software in other forms. For your convenience, other groups that are helping to spread GNU software are listed below. Please note that the Free Software Foundation is not affiliated with them in any way, and is not responsible for either the currency of their versions or the swiftness of their responses.

These Internet sites have some GNU programs available for anonymous FTP:

`louie.udel.edu`

`scam.berkeley.edu`
`spam.istc.sri.com`
`bu-it.bu.edu`
`uunet.uu.net`
`nic.nyser.net`
`wsmr-simtel20.army.mil` (under
 'PD:<UNIX.GNU>')
 and `cc.utah.edu` (VMS GNU
 Emacs).

Those on the SPAN network can ask
`rdss::corbet`

Information on how to obtain some GNU programs using UUCP is available via electronic mail from:

`hao!sciocom!getzal!upba!ugn!nepa!denny`
`arnold@skeeve.UUCP`
`uunet!hutch!barber`
`hqda-ai!merlin`
`acornrc!bob`
`ames!killer!wisner`
`mit-eddie!bloom-`
`beacon!ht!spt!gz`
`sun!nosun!illian!darylm`
 or `postmaster@uunet.uu.net`

Ohio State also makes GNU programs available via UUCP. They post their instructions monthly to newsgroup `comp.sources.don` USENET. Current details from Karl Kleinpaste `karl@tut.cis.ohio-state.edu` or `...!osu-cis!karl`; or Bob Sutterfield (substitute bob for karl in the above addresses).

Information on obtaining floppy disks of GNU Emacs for the AT&T Unix PC (aka

3B1 or PC7300) is available via electronic mail from: `brant@manta.uucp`

Thank GNUs

Thanks to all those mentioned in GNUs-worthy Flashes and the GNU Project Status Report.

Thanks to Hewlett-Packard for their very large cash donation and two Spectrum workstations.

Thanks to the MIT Laboratory for Computer Science, and its director, Professor **Michael Dertouzos**. LCS has provided FSF with the loan of a Microvax for program development.

Thanks to the MIT Artificial Intelligence Laboratory for invaluable assistance of many kinds.

Thanks to **Dr. T. Smith**, **Dave Probert**, and the CS Department at UCSB for giving GNU staffer **Brian Fox** resources and space, and special personal thanks from **Brian** to **Matt Wette** for invaluable aid and support.

Thanks to **Brewster Kahle** of Thinking Machines, Inc. for the loan of a Sun 4/110.

Thanks to **K. Richard Magill** for his donation of an AT&T Unix PC.

Thanks to **Arnold Robbins** and **Dave Trueman** for their work on GAWK and the GAWK manual.

Thanks to **Barry Kleinman** and **Andre Mesarovic** of Index Technology for copying Sun cartridge tapes and to **Mark Nahabedian** of Phoenix Technologies Ltd. for copying Sun cartridge tapes at the 11th hour.

Thanks to **John Klensin** of the INFOODS Project at MIT for making our VMS master tapes.

Thanks to Sony Corp. and to Software Research Associates, Inc., both of Tokyo, for sending us Sony workstations. SRA has also given us a cash donation and lent us a full-time staff programmer and a technical writer.

Thanks to NeXT, Inc., for their cash donation.

Thanks to the Mach Project in the Department of Computer Science at Carnegie Mellon University, for lending us a Sun 3/60 and 300 MB disk drive.

Thanks to all those who have contributed ports and extensions, as well as those who have contributed other source code, documentation, and good bug reports.

Thanks to those who sent money and offered help. Thanks also to those who support us by ordering Emacs manuals and distribution tapes.

The creation of this bulletin is our way of thanking all who have expressed interest in what we are doing.

Order Form 17 January 1989

This form is effective January 1989-June 1989.

All software and publications are distributed with permission to copy and redistribute.

_____ \$150 GNU Emacs and other software, for Unix systems, on 1600bpi reel-to-reel 9-track tape in tar format.

The tape also contains:

- * MIT Scheme (a dialect of Lisp)
- * T, Yale's implementation of Scheme
- * Bison (a free, compatible replacement for yacc)
- * Hack (a rogue-like game)
- * GNU Chess (a chess playing program with an interface to X).
- * GDB (the GNU source-level C debugger)
- * The X window system (a window system for bitmap displays written at MIT) (version 10r4)

_____ \$150 GNU Beta Test tape, for Unix systems, on tape as in last item.

The tape includes:

- * GCC (the GNU C Compiler)
- * G++ (the C++ front end to GCC)
- * Bison (a free, compatible replacement for yacc)
- * Flex (Vern Paxson fast rewrite of lex)
- * Gnostscript (a Postscript interpreter)
- * Gawk (the GNU implementation of the AWK programming language)
- * Gas (the GNU Assembler)
- * Gnuplot (an interactive mathematical plotting program)

* GNU object file utilities (ar, ld, make, gprof, size, nm, strip, ranlib, et al.)

* other GNU utilities (diff, grep, tar, et al.)

* the freed files from the 4.3BSD-Tahoe distribution

_____ \$150 MIT X Window System X11R3, for Unix systems, on tape as in last item.

_____ \$175 GNU Emacs and other software, for Suns, as in the first item, on DC300XLP 1/4inch cartridge tape, in QIC-24 format.

_____ \$175 GNU Beta Test tape, for Suns, as in the second item, on tape as in last item.

_____ \$175 MIT X Window System X11R3, as in the third item, on tape as in last item.

_____ \$150 GNU Emacs source code and binaries, for VMS, on 1600 bpi reel-to-reel 9-track tape in VMS backup format.

_____ \$150 GNU C compiler source code and binaries, for VMS, on tape as in last item. Includes Bison and GAS.

_____ \$15 GNU Emacs manual, ~300 pages. The manual is phototypeset, offset printed, and spiral bound; it includes a reference card.

Thus, a 1600 bpi tape and one Emacs manual come to \$165.

_____ \$60 Box of six GNU Emacs manuals, each with reference card.

_____ \$1 One GNU Emacs reference card, without the manual.

_____ \$5 Packet of ten GNU Emacs reference cards.

_____ \$10 GDB Manual, ~70 pages, side stapled.

_____ \$10 Texinfo Manual, ~100 pages, side stapled. Texinfo is GNU's structured documentation system, included with GNU Emacs. This manual describes how to write Texinfo documents.

_____ \$10 Termcap Manual, ~60 pages, side stapled. Documents the termcap library and GNU's extensions to it. The GNU termcap library is included with GNU Emacs.

_____ \$10 Bison Manual, ~70 pages, side stapled.

_____ \$10 Gawk Manual, ~100 pages, side stapled.

_____ \$10 Make Manual, ~100 pages, side stapled.

Prices and contents of tapes are subject to change without notice.

_____ If ordering from Massachusetts: add 5% sales tax.

_____ If outside of North America

and Hawaii: for shipping costs, please add \$60 for each box of Emacs manuals; for tapes or unboxed manuals, please add \$15, and then add \$15 more for each tape or unboxed manual.

_____ Optional tax deductible donation.

Orders are filled upon receipt of check or money order. We do not have the staff to handle the billing of unpaid orders. Please help keep our lives simple by including your payment with your order.

Make checks payable to Free Software Foundation.

All software from the Free Software Foundation is provided on an "as is" basis, with no warranty of any kind.

GNU's Bulletin

Copyright (C) 1989 Free Software Foundation, Inc.

675 Massachusetts Ave

Cambridge, MA 02139, USA

Telephone: (617) 876-3296

Electronic mail: gnu@@prep.ai.mit.edu

Permission is granted to anyone to make or distribute verbatim copies of this document as received, in any medium, provided that the copyright notice and permission notice are preserved, and that the distributor grants the recipient permission for further redistribution as permitted by this notice.