

TR-O-0072

40

光電波通信研究所における
UNIXを用いた計算機環境とその運用

竹内 和則 藤井 智史 松井 渉

1994. 3. 31

ATR光電波通信研究所

光電波通信研究所における UNIX を用いた計算機の環境とその運用

無線通信第一研究室 竹内 和則

無線通信第二研究室 藤井 智史

無線通信第二研究室 松井 渉

The configuration and operation of the computers and its networks based on unix system in ATR Optical and
Radio Communications Research Laboratories

Kazunori TAKEUCHI , Satoshi FUJII and Wataru MATSUI
, Researchers

(株)ATR 光電波通信研究所、京都府

ATR Optical and Radio Communications Reserach Laboratories, Kyoto-fu 619-02 Japan

著者指定: 赤表紙

連絡先:

竹内和則 ATR 光電波通信研究所 無線通信第一研究室

〒619-02 京都府相楽郡精華町光台2丁目2

TEL 07749 5 1511(1520)

目次

1	まえがき	3
2	運用コンセプト	4
2.1	user の要求	4
2.2	無駄な CPU 消費の回避	4
2.3	CAD 価格が CPU 性能に正比例することの利用	4
2.4	運用管理側からの要求	5
3	Hardware 構成	6
3.1	共用系構成	6
3.2	個人管理 work station 構成	8
3.3	LAN 構成と server の console 端末接続	10
4	Software の構成	11
4.1	Sun SPARC Server & Station の disk の内容	11
4.2	Sun SPARC Server & Station の disk の OS	11
4.3	Solaris 2.x(SunOS 5.x) との対比	12
4.4	SunOS 4.1.3+JLE との比較	12
4.5	NFS client と server	13
4.6	Sun 上で動作する市販 software	13
4.7	Sun SPARC 上で操作する Free Software	15
4.7.1	光電波での Free Software 概要	15
4.7.2	光電波における Free Software 導入手順	15
4.7.3	Free Software で必要とされる使用許諾が必要な software	16
4.7.4	Free Software の管理	17
4.8	L ^A T _E X による論文作成	18
4.9	NFS による user data の local 管理	19
4.9.1	概略	19
4.9.2	NFS 管理における file 配置の手法と優劣	19
4.9.3	光電波の NFS 管理における file 配置	20
4.9.4	automount の利用	20
4.9.5	TITAN, CONVEX の rs?? からの automount	21
4.9.6	NFS Clinet rs?? と rs?? 同士の NFS	21

4.9.7 NFS Client rs?? と rs?? 同士の login の対策の現状	22
4.9.8 automount によって引き起こされる障害	22
4.10 光電波で用いられる通常の数値計算、出力環境	23
4.11 図形出力の file 形式	26
5 購入時の事例	29
6 支援作業	32
7 まとめ	35
8 謝辞	36
9 Appendix-I	37
10 Appendix-II	40

1 まえがき

本テクニカルレポートは、Sun WorkStation 上で *Mule 1.1* 及び \LaTeX を用いて作成された。この編集組み版 system は Macintosh 上の PageMaker や日本語 Word Processor のように画面上での優れた GUI は与えないが自由度や、一旦決定したスタイルの移植性は圧倒的に高い。また基本的に *Free software* である。

光電波でも \LaTeX や \TeX の user は多く、いくつもの論文やレポートが \TeX を介して作成されている。また、多くの海外や国内の学会で \TeX format の原稿を Email で受け付けることが増えてきた。

また、本文中の図面は *idraw*, *tgif* を用いて作成されている。これらは \TeX 同様に広く WorkStation で用いられ OHP(Over Head Projector) sheet の作成や論文中の図版作成に用いられている。

Workstation 上で上記の \TeX や描画 software を良好に動作させるにはいくつもの system 上の条件が揃うことが必要であり、簡単とは言えない。しかし、一旦 service in するとその利点は Fortran77 や C の出力結果をそのまま貼り付けられることや、OHP 作成用に使用していた *idraw* 等の図表をそのまま論文に用いる図面に電子的に使用出来る、Email の内容をそのまま論文に copy したり \TeX 原稿をそのまま Email に貼り付ける等が可能等枚挙にいとまがない。これは \LaTeX が基本的に plain text と *eps(encapsulated postscript)* に限っているために unix との親和性が高いため得られる利点である。

本レポートは、単に優れた user interface としての Xwindow を有する unix machine としての WorkStation ではなく、*Freeware* を多く導入することにより「優れた文房具」としての機能も果たす WorkStation を光電波に実現し、WorkStation から一步も動く事なくシミュレーションや実験結果の整理とグラフ作成を行なえ、それらをそのまま用いて論文作成までを可能にする環境を紹介する。加えてそれらを実現する際に必要な知識の内、光電波固有の問題で、一般の書籍雑誌にも書き下だせないような子細な情報までを記載することにより光電波の研究補助の助けになることを目的としている。

2 運用コンセプト

2.1 user の要求

計算機利用者に対して提供すべきサービスは次の2点に絞られる。

- (1) user friendly な環境
- (2) 高速な演算能力 (CPU のみでなく compiler の性能も含めて)

このため、光電波では

- (1) として安価な (約 300 万円) Sun workstation や X 端末等の Xwindow system (必ずしも GUI が常に user friendly とは限らないが)
- (2) として *CONVEX C240, TITAN 3000* 等の演算専用機

を用意して2段構えの構成を用いている。

2.2 無駄な CPU 消費の回避

このとき CPU を interactive に消費するため演算の妨げとなるエディターが一番の問題となる。すなわち光電波で一番使用されている editor NEmacs は lisp 処理系の editor であるため CPU の消費は無視出来ない。(2) の CPU 上で NEmacs を使用すると CPU の消費量に比べて得るものは少ない。このため (2) の CPU 上で使用する file の edit は (1) の CPU で可能なように NFS を張り、(1) の CPU が専ら edit に使用されるようにしている。

上記は設定の難しい NEmacs のための個人環境の整備にも有利である。すなわち、NEmacs は個人の使用する command 群や delete 等の鍵配置の都合に合わせて動作できるように \$HOME/.emacs file を用意しており、各個人の好みに合わせてそれら設定できるが、全部で3種の machine (*Sun, TITAN, CONVEX*) の Hardware の違いからくる細かい部分をそれぞれ設定するのは徒労である。*Sun* 用の .emacs さえ設定してしまえば上記3台の machine 上の file はその *Sun* の NEmacs で edit でき、常に一番使い易い editor を使用できる。

2.3 CAD 価格が CPU 性能に正比例することの利用

市販 software の価格体制に対応するためにも *Sun* と *TITAN, CONVEX* という2段構えの構成が有利である。演算結果の graph 出力等には CAD や描画 software が必須であるが、これを *TITAN* や *CONVEX* 用に用意すると高価なものとなる。同一パッケージの software も *Sun* 用と *CONVEX* 用で大きく価格が異なるからである。このため、*TITAN, CONVEX* では AVS や IMSL 等 CPU の能力が必要な CAD や Library を除いては一切の optional software や third party の Package software を載せず、専らそのような software は *Sun* 上に載せることにより NFS でもって処理することで対応している。

2.4 運用管理側からの要求

前節までで述べたユーザーの要求と諸般の現実的な問題に加えて計算機管理の立場からは以下の要求がある

(3) 管理のしやすい画一的な環境かつ一括管理が可能な環境

これは本来 user の自由度を奪うものだが、自由度を求める程の user は十分 skillful であり、画一的環境内で自前で自由度を上げる能力を持つため、光電波では (3) のポリシーで運用を行なっても問題が少ないと考えてよい。(3) は NIS(Network Information System:Yellow Page Service) や NFS(Network File System) を積極的に採用することにより得られる。[1] [2]

3 Hardware 構成

3.1 共用系構成

- rs04 : *Sun SPARC Server 670MP*

CPU SPARC 40MHz*2

RAM 128MB (70nsec 4MB SIMM*32)

IPI DISK 1GB*2 ベデスタル内蔵

SCSI DISK 1GB2, 2GB*1 LUNCH BOX type 外付け disk

(a) IPI は system boot up 等基幹 disk として用いており SCSI は Storage 的な使用法をしている。

(b) IPI disk は disk interface としての IPI 規格が SCSI に押されたため商業的に劣勢になった。また IPI は専用ベデスタル (MASS STORAGE PEDESTAL) に内蔵して供給されてきたが、その筐体の熱放散処理が設計ミスでうまくいかず多くの site で disk crash をまねいたため現在 Sun の Price List から外れている (販売中止)。現在の光電波の IPI disk はベデスタルのファンの交換や disk 自体の lot 不良に対応する交換を重ねており、1993 春-1994 春までの期間はその動作は安定している。

(c) *SPARC Server Series 6XX* は現在既に Price List から外れているが、CPU の SPARC 40MHz から Super SPARC 50Hz 等への Upgrade path は残されており、rs04 の強化は Mbus 上の CPU module 交換で簡単に可能である。

(d) 当初 SWAP disk “id000b” は 98.4MB を割り当てていたが、幾つか process を走らせて試行したところ、128MB の RAM が十分に生かせず SWAP disk 容量の 98MB 辺りで頭打ちになっていることが明らかになった。このため、disk 構成をあらため SWAP を 256MB に拡張して対処している。

(e) rs04 は rs01 と同じく、/dev/console に VT220 terminal+printer を接続し、/etc/rc, /etc/rc.local のエラー出力、標準出力を /dev/console に吐かせることによりプリンター上にシステムの状態をモニターしている。VT284 でなく VT220 を使用しているのは VT220 の Printer Port 出力は Printer の不調時に対応できるが VT284 は Printer 不調時に terminal 自身まで keyboard lock されてしまい printer trouble による CPU 障害という事態を引き起こす可能性が大きいからである。

- rs01 : *Sun SPARC Server 630MP*

CPU Super SPARC40MHz*2

RAM 128MB (70nsec 4MB SIMM*32)

SCSI DISK Part1 1GB*1 本体内蔵 SCSI

SCSI DISK Part2 1GB*4 ベテスタル内蔵

- (a) SCSI part1 は、CPU board に付属の SCSI port で 1 台の SCSI disk “sd0” のみを接続している。これは “sd0” が /export すなわち NFS client が常に要求する OS に付属の /usr 等部分を有するため一番 access が盛んな disk であり、スループットの向上の為他の disk を接続させないことを目的としている。この access 頻度は /bin/iostat, /usr/etc/nfsstat で知ることが可能である。
- (b) SCSI part2 は Sbus Card 上にある SCSI port に接続される、同 card はさらに 2nd ethernet port も具備しているがそれは使用していない。
- (c) SCSI disk ベダスタルは合計 8 台の 3.5” Hard disk の収納が可能であり、現在 4 台使用しているの、あと 4 台の増設が可能である。ただし、Sun Microsystem は 1 系統の SCSI に Hard disk の接続を 4 台までしか推奨しておらず、新たな disk 増設にはさらに FAST SCSI Sbus Card の増設が必要である。

- atr-rd : *Sun SPARC Station 2Plus*

CPU Weitek Power μ P 80MHz

RAM 64MB

SCSI DISK 2GB*1 + 0.4GB*1 内蔵

- (a) atr-rd は旧 atr-rd (DEC5810) に比してその機能を大幅に SPARC Server (rs01,rs04) に移管している。このため能力的には他の Server machine に比べて性能的に劣る Hardware となっている。基本的には以下の機能をはたすと考えてよい。

- (a-1) Sendmail (smtp)

- (a-2) INN (nntp)

- (a-3) ftp (ATR 外部の ip-reachable な site と)

- rtitan01 : *TITAN 3000*

CPU MIPS R3000 *3

RAM 64MB+128MB

SCSI DISK 300MB*3

- (a) disk 容量ぎりぎりのシステムなので、CONVEX の disk を NFS mount することにより user area の不足分を補っている。
- (b) 3CPU を Parallel に実行できるため 40MFLOPS の性能を持ち、中規模の計算に適している。
- (c) Swap space が 60MB しかないが、SUN と異なり約 160MB 程度の job であれば十分こなす。
- (d) SCSI interface を備えるが、Connector が独自のもので、かつ接続できる disk も Maxtor 社以下の型番のもののみが推奨されている。それ以外は Kubota 社の TAHITI MO drive に限られており、Sun WS のそれに比べて disk の増設は困難である。1994 年春現在 Sun 用 SCSI disk は 1GB で 200Kyen を切る実勢価格となっていることを考えると NFS による柔軟な運用が妥当と思われる。
 - XT-8380SH 380MB 520Kyen
 - XT-8760SH 760MB 970Kyen
 - XT-81600SH 1.6GB 1490Kyen

- (e) /tmp は重要な directory であり compile 等では時に大きな容量が必要となるがこの directory は NFS による拡張が不可能なので各 user は環境変数 TMPDIR を \$HOME/tmp/ に設定することにより system disk の disk full を回避している。

- atr-dv : CONVEX C240

CPU CONVEX C240 4-CPU

RAM 512MB

DISK 8GB (8 台の stripping により高速化を図っている)

- (a) Swap に 1.5GB を割り当て、大規模長時間計算に十分に対応できるよう配慮されている。
- (b) 4CPU が parallel,vector 化されて走行できるので、一つの process を 4 つに thread に分割し最高で 1CPU の 4 倍の速さで高速に実行できる。
- (c) 8GB の高速な disk を有するが、大変高価であるので、必要性の低い file は安価な Sun の disk 等に移動させている。

3.2 個人管理 work station 構成

- rs?? : Sun SPARC Station(Sun4 series)

CPU SPARC V.7 ないし V.8 の命令系を有する

RAM 16MB-128MB

DISK 200MB-4GB Appendix-II に光電波の 94 春現在の一覧を示す。

- (a) Keyboard は Type IV Unix ないし Type V Unix Keyboard を指定して購入することとして Ascii 配列を採用し操作性や他のマシンとの融和性を重視。
- (b) Graphic Card は CG3,GX,TurboGXplus 等 X11R5 の Sample Server が対応できるカードにしている。ZX,SX 等は高速高価であるが、それ専用の Server software が必要であり、現在使用中の SunOS 4.1.3 上では十分に利用できないか support されない。
- (c) 全て内蔵 Floppy disk drive を有しており、IBM-PC format の 2HD,2DD drive に対応する software(mttools,dosf) により、IBM-PC,Macintosh,PC-9801 等とのデータ交換を容易にしている。
- (d) ほとんど全ての rs?? は dataless 構成である。このため Server である rs01 の設定の変更、file の追加だけで新しい software の実行が可能となる。
- (e) いくつかの machine ではその Sbus 上に IEEE488.2 bus interface を具備し、計測に対しても用いられるようになっている。
- (f) CPU 性能を客観的に示すことは困難で、各種 Benchmark test といえども客観的な数字を総合的には示せないがある程度の目安は得られる。次回に市販 WorkStation の性能を spec 値で揚げる。

SPEC rate and FLOPS of processor/machine (data from Catalog)

Machine Name [CPU Clock]	SPECint'92	SPECfp'92	MFLOPS
SPARC Station1+/IPC [25MHz]	13.8	11.1	1.8
SPARC Station Classic [50MHz]	26.4	21.0	?
SPARC Station2 & IPX[40MHz]	21.8	22.8(21.5)	4.2
SPARC Station2+(Power uP) [80MHz]	32.2	31.1	33.3(Peak)
SPARC Station Classic70 [70MHz]	50	43	?
SPARC Station10/30 [36MHz]	44.2	52.9	10.6
SPARC Station10/41 [40MHz]	52.6	64.7	17.2
SPARC Station10/52 [50MHz]	58.1	71.4	19.0
SPARC Station20 [60MHz]	78	100	?
HP9000/712 [60MHz]	58.1	79	40
HP9000/735 [99MHz]	80	150	40
HP9000/735 [125MHz]	136	201	?
DEC3000/300 [150MHz]	66.2	91.5	24.5
DEC3000/500 [150MHz]	74.3	125.1	30.1
DEC3000/600 [175MHz]	105.3	162.1	35.0
i486DX2 [66MHz]	32.2	16.0	?
i486DX4 [99MHz]	45.7	22.2	?
Pentium [66MHz]	67.4	63.6	?
MC68040 [40MHz]	33	24	?
PowerPC 601 [66MHz]	60	80	?
PowerPC 601 [80MHz]	72	95	?
PowerPC 603 [66MHz]	60	75	?
PowerPC 603 [80MHz]	70	80	?
CONVEX Exemplar [99MHz]	?	?	25G/128chips
Fujitsu AP1000 [50MHz]	?	?	51.2G/1024chips
CRAY T3D [150MHz]	?	?	372G/2048chips

3.3 LAN 構成と server の console 端末接続

光電波における LAN の構成を図 1 にしめす。図中 rs?? 等の NFS client や Macintosh 等は rs01,rs04 にしか依存していないように見えるが、実際は Email や news 等を atr-rd に、rs?? は rs04 上の NIS に依存しているしまた jserver も rs04 上のを使用している。Macintosh user も Mathematica の server を rs01 に依存しており実際の関係は複雑である。

また、rs01,rs04 の配線接続を図 2 に示す。rs01,rs04 は Workstation と異なり keyboard や CRT が bus から直接接続されないので EIA232c を介して console が接続され、作業記録や error message が取れるよう常時 console printer が接続されている。

図 1: LAN の構成

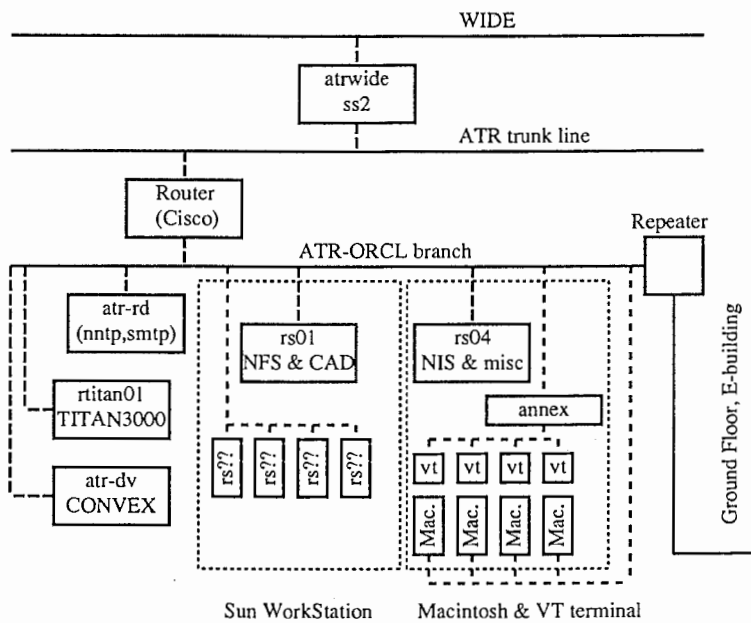
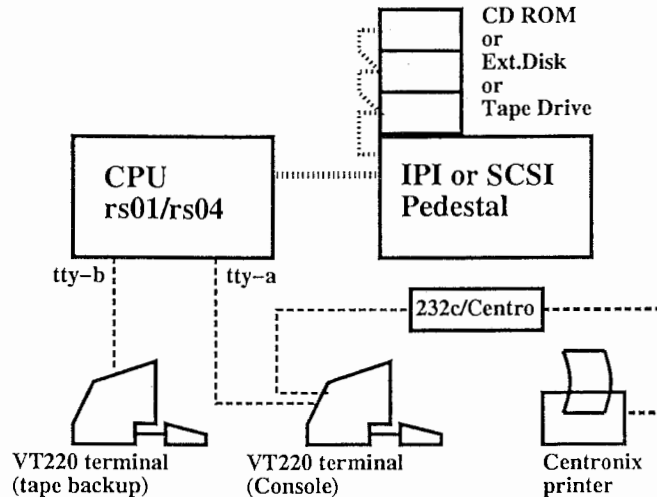


図 2: SCSI と console 端末の接続



4 Software の構成

4.1 Sun SPARC Server & Station の disk の内容

rs12,rs16,rs27 を除くほぼ全ての SPARC Station は SPARC Server rs01 の NFS Client として dataless system となっている。すなわち自前の disk には /etc,/sbin,/usr2/home??./tmp 等の directory をもち、他の主要な directory である /usr 等は Server に依存し NFS mount している。また /etc 以下の file もシステムの構成上 Server と同じであることが要求される部分すなわち password, group, hosts, services, netmask, network 等も NIS を用いて Server で一括管理されている。このため新しい rs?? が加わってもその IP address や hostname 等の情報は Server の主 file を書き換えて NIS の date base を書き換えるだけで全 rs?? の情報として更新される。

4.2 Sun SPARC Server & Station の disk の OS

OS は SunOS 4.1.3 を用いており、いくつか Patch をあてている。

100173-10, NFS Jumbo Patch

100377-05, Sendmail Jumbo Patch

100623-03, UFS Jumbo Patch

100444-48, Open window 用 Patch

100726-11,-12 SS10,600MP(sun4m architecture) 用

100170-10, /bin/ld 用 Patch

100689-01, Kernel 用 object

また現在 SunOS 4.1.3-U1 が出荷されている。これは主たる Patch が既にあたっているものであり、kernel reconfiguration が必要になったときはこちらを使用することが望ましい。

さらに、security 問題への対策¹として IP reachable な machine に必ず施さなくてはならない patch として以下が知られている。

100478-01, /usr/openwin/bin/xlock

100296-04, /usr/etc/rpc.mountd

100507-04, kernel reconfig

100372-02, /usr/etc/tfsd

100103-11, file permission change from shell script

100567-04, kernel

100564-05, /usr/etc/rpc.yppasswdd, /usr/etc/pwdauthd

100482-04, /usr/etc/yppserv, /usr/etc/yypxfrd, /usr/etc/portmap

100513-02, kernel

100623-03, kernel

¹Message-ID (1994Mar22.094409.12766@riksun.riken.go.jp) jp.inet.security.announce 22MAR94

100173-10, kernel
100383-06, /usr/ucb/rdist
100452-28, /usr/openwin/lib/libxview.a
100305-11, /usr/lib/lpd
100891-01, /usr/lib/libc.a
100224-06, /bin/mail /bin/rmail
101080-01, /usr/lib/expreserve
100448-02, /usr/openwin/bin/loadmodule
101200-02, /usr/kvm/modload
100377-08, /usr/lib/sendmail
100593-03, /usr/etc/dump
100272-07, /usr/etc/in.comsat
101480-01, /usr/etc/in.talkd
101481-01, /usr/etc/shutdown
100909-02, /usr/etc/syslogd
101482-01, /usr/bin/write

4.3 Solaris 2.x(SunOS 5.x) との対比

現在光電波で使用している Server Machine rs01,rs04 はいづれも 2CPU 構成である。Sun Microsystem 及び Sun Soft では 2CPU 構成の Machine に対しては SunOS 5.0 以上を奨めているが、これは SunOS 5.0 以上では process をいくつかの thread に分割し、それらの thread を複数の CPU で同時に処理させることにより process の処理速度を上げられるということを根拠としている。しかし光電波では process 分割による高速化は rtitan01 や atr-dv に専ら任され rs01,rs04 は安定性を求められているため、OS として安定しており bug 情報も充実している SunOS 4.1.3 を用いている。

2CPU 構成の SunOS 4.1.3 は、process は分割しないが、process を 2 つの CPU に振り分けることは行なっており実際上は全く問題ない。尚 rs01,rs04 (630MP,670MP) は Mbus スロット数からみて 3CPU 以上が実装可能ではあるが、SunOS 4.1.3 で 3CPU を動作させた実績は vendor にも報告されておらず、また、Mbus CPU module が 3 枚物理的に入るか否かも不明であり、現在は 2CPU で使用している。

4.4 SunOS 4.1.3+JLE との比較

光電波では JLE(日本語環境)を加えた SunOS4.1.3+JLE は使用していない。これは JLE を導入すると

- DNS[3] が使用出来ない。このため Sendmail-MX も使用出来ない。

- 日本語化による bug の混入が著しい時期があり、信頼性において純正 4.1.3 より劣る等弊害が多い
- 日本語は Wnn ないし Canna System 等で入力できる。また左記 2 者の方が辞書や取り扱い情報が充実しており、また各種 utility との共用が簡便に設定出来統一した日本語入力 method が user に提供出来る。[4] [5]
- 日本語 manual や 8bit 透過のフィルターツールは /usr/local/bin/ 以下の Free Software を使用することで間に合う。
- 日本語用と native の dynamic link library がまとめて一つの file に入っているため native な関数を使用する際 locale により切り換えなくては行けないが、この切り換え用環境変数 LANG で致命的な動作の違いを生むので使用者が全員 JLE 動作について細かい知識を要求される。
- JLE では JLE 用の Patch 及び純正 OS 用 Patch の二種をうまく使い分けないと行けない等管理側の負荷が重い。
- XPEG 等の Library は JLE 下では使用できないか再コンパイルが必要である。
- 日本語 keyboard を使用しなくては行けないがこの日本語 keyboard の操作性が NEmacs やその他 utility と合わずユーザインターフェースが著しく劣る。

等、利点が全く無く欠点が多いとの観点から使用を止めている。

実際 JLE が絶対に必要となるのは HP-PE/ME10 等一部の CAD Software のみであり JLE の必要性は殆どないと考えてよい。

4.5 NFS client と server

rs?? は Server Machine の rs01:/export/exec/sun4.sunos.4.1.3 を /usr として NFS mount することにより /usr を共有する。SPARC chip は現在光電波において SPARC12.5MHz, 25MHz, 40MHz, Super SPARC 36MHz, 40MHz, 50MHz 版の 6 種があるが前 3 種が Sun4c, 後 3 種が Sun4m という architecture を有し、/bin/uname -a, /usr/etc/showrev, /u 等で区別できる。SPARC のバージョンでいえば前 3 種が version7, 後 3 種が version8 と呼ばれる。これ等は別々の kernel を用意しなくては行けないので kernel の reconfiguration 用 directory である /usr/kvm は machine により Sun4c 用と Sun4m 用を分けて用意している。実際 server 上で共有される /usr 自体は version8 と binary compatible な version7 のコードとなっている。

4.6 Sun 上で動作する市販 software

実際に光電波にある Sun 上の市販の package を示す。

- f77-2.0, acc 等 SunSoft 純正コンパイラ
- CA-Disspla version11 数値グラフ作成サブルーチン群

(c) IMSL, C/MATH(IMSL for C-language) 数値計算用関数サブルーチン群

(d) Mathematica 関数の直接表現形式による直接計算

(e) Spyglass 3次元グラフ描画 CAD

(f) HP-MDS マイクロ波シミュレーター

(g) Explorer 2. 5次元マイクロ波回路解析シミュレーター

(a) 以外は用途の特化された専用 software が多く、一般的に使われる細かい tools は全て Free Software に依っている。(a) にしても F.S.F. の gcc(GNU C compiler) に比べて performance が悪いという事例の報告も少なくない。

4.7 Sun SPARC 上で操作する Free Software

4.7.1 光電波での Free Software 概要

光電波では主として使用する software のほとんどが Free Software である。Free Software は Public Domain Software とは同義ではない。license to use は Free であり、source code の改変も自由であるが幾つかの制約は存在する。最も有名な例として *Copy Left* を掲げている *Free Software Foundation* の「製品」を上げる。Appendix-I に GNU products (GNU Emacs) の著作権に関する条文を挙げる。光電波の代表的な Free Software では以下が愛用されている。

(a) X11R5 の contrib に含まれる package

(b) F.S.F. の GNU Emacs 18.59 ないし Mule 1.1

(c) /usr/local/bin/ ないし /usr/X11R5/bin/ に備わった software

GNU plot, Xgraph, idraw, ghost script, ghost view, jL^AT_EX, kinput2 等

これらは概して市販の package software に比して十分強力であり、また保守に相当する patch の流布も非常に早く、Internet でのボランティア投稿者による support も充実しており有料の市販 software に比して無料であるということに起因する手抜きや不完全さは見受けられない。よって Free Software は license 管理の手間が無い分だけでも有利であり、大いに活用すべきと考えられる。

Virus の危惧は基本的には以下のように杞憂である場合がほとんどである。

- その Source code と software の配布元とが常に明らかなので software 中に Virus を混入させづらいこと
- archive されているものなら Internet で広く流布されてからの月日もたっており、もしも virus が含有されているなら既に fj.sources.d 等 Internet の news group で報告が上がること
- user の階層管理の概念のない PC-DOS, Macintosh 等と異なり、/etc/chmod +s を加えた executable でない限り実行ファイル自体を使った「トロイの木馬」による root 特権が容易には得にくいこと
- Virus よりも IP reachable な機能を生かして接近する方が hacker には興味深いであろうと考えられること

fj の多くの Site でも IBM-PC か MACINTOSH 用 PDS では警告がでているが UNIX の Free Software Sources ではみかけない。

4.7.2 光電波における Free Software 導入手順

これらの Free software を中心とする光電波のシステムの構築順序を以下に示す。

(a) Vender から購入した SunOS 4.1.3 ないし 4.1.3_U1 の install

- (b) patch 100???-?? を施し、vmunix の再 configuration。このとき Server は max users を 64 へ拡張し、Clients は不要な devices を切り落とすことにより performance の向上を図る。
- (c) /usr/local/bin/patch の install。この際の make には /bin/cc を使用
- (d) /usr/local/bin/sed,diff,gunzip,bison 等 Free Software の make に最小限必要な tool の install (これらは Free Software の makefile の際に有効である)
- (e) gcc の install、gcc は Free Software の install には欠かせない C コンパイラで非常に強力である。F.S.F. の GNU project による代表的な「製品」である。
- (f) X11R5 の mit 部分の install。Sun が SunOS 4.1.3 用として用意するところの Xwindow system である Open Window は X11R4 であり mit 部分も一つ前の版ある。このため日本語の取り扱いを始め不備が多い。X11 の mit 部分は多くの主要 vendor で一部を省いて供給するので、特に Free Software の make に必要な各種 library 等は供給を拒まれる。このため SunOS 4.1.3 上でそれら代表的な library を含んだ X11R5 を make world する。この make world で gcc を用いた場合、/bin/cc による make に比べて 20% 程高速化される。
- (g) tcsh の make 及び install tcsh は最も光電波で愛用されている拡張 C-shell である。
- (h) Wnn の make 及び install、Wnn 自体は X11R5/contrib/Wnn として全体が供給されるが X11 を使用するしな
いに関わらず必要となる標準的な日本語かな漢字変換機能である。この Wnn は多くの Free Software の日本語
変換に必要なだけでなく、JLE と比して岩波国語辞書等が市販されたり、市販のワープロソフトウェアである
D P 図形等の日本語変換部分に必要であったりするためその導入するメリットは非常に大きい。
- (i) Mule,NEmacs の make/install
- (k) 上記以外の各種 Free Software の make/install
- (j) jman の整備 (catman から作成)

4.7.3 Free Software で必要とされる使用許諾が必要な software

上記 Free Software は全て source code で供給され user の好みに合わせて書き換え可能であり、license の問題が無いと予算や license 管理上大変有利であるが、いくつかの Software は有料な市販 software の一部ないし全部を利用しているのでこの点に気をつけて使用しなくてはならない。特に Free であるとの先入観から有料部分を含めて全て複製して Local Machine で使用する可能性が大きく、適法運用のためには良く知り、見分ける必要がある。以下に示すものは Free Software で利用されている有料市販 software である。

- (a) 大日本印刷の NTT j_TE_X 用 Fonts (j_TE_X 自体は Free)
- (b) Wnn 用岩波国語辞典辞書 (Wnn 自体は Free)
- (c) Sun 上で動作する Postscript tool 群 enscript

4.7.4 Free Software の管理

光電波での計算機管理の大きな仕事の一つにこの Free Software の make/install/patch/version up がある。これは debug の情報公開が早く、機能強化の著しい Free Software において必要な maintenance 作業である。これ等の拡張によって得られるものはみかけ上大きくないが、操作上の掻痒感の減少や、作成された file の汎用性の拡大等底辺を押し上げるものとして重要である。

市販の software の場合あからさまに目に見える機能拡張や目新しい GUI の導入等が version up につきものであるが、debug 等は少なく、また高速化も行なわれることが少ない。このためひたすら重い software となり、必ずしも user の望むものではないことが多い。通常光電波での Free Software install はその必要な情報が fj 等に十分流れてのち行なっている。これは install 作業の工数を減らすために必要な期間である。Free Software は通常 gcc を用いて make することを前提としているので、gcc の maintenance と bug 情報は Free Software の make には必須である。光電波では fj.sources 等 fj や usenet に流布する software は原則として全て保存し、また CD-ROM 等でも用意することで自前の archive を充実させるとともに、xarchie や anonymous ftp 等で外部 site にある流布されない software の入手も必要に応じて行なっている。

4.8 L^AT_EX による論文作成

光電波で最も使われている WorkStation 用 Software 群は L^AT_EX [6] [7] [8] を中心とした論文作成時に最大の効力を発揮する。T_EX は Macintosh 上でも動作するが (pT_EX)、Object を対象として操作する Macintosh OS 上より文字列 (ASCII, JIS, EUC, SJIS) を純粹に扱う tool 群の揃った UNIX 上の方が L^AT_EX 向きといえよう。以下に L^AT_EX による論文作成の実際の手順を示す。

(a) jBIBT_EX による参考文献 database の作成

(b) idraw, tgit, xfig, xgraph, displa-11 等による eps (encapslated postscript) 形式の図表の作成

(c) NEmacs3.3.2 ないし Mule 1.1 + cmutex.el [9] による jL^AT_EX 統合環境

- 文書入力
- 英語論文作成時に必要な online dictionary 利用
- spell checker 利用
- thesaurus 利用
- T_EX コンパイル
- previewer による印刷イメージでの検査 (xdvi, texx2)

(d) 長文の場合、俗にいう「流し込み」を実施している場合も多く、その際は Personal Computer, Word Processor 等からの文書交換に dosf ないし pc-ftp、日本語文字コード変換に nkf を使用する。

(e) dvi2ps を用いた postscript 形式への変換

(f) ghostscript を用いた postscript preview

(g) lpdaemon を用いた printer 制御

L^AT_EX 使用の際に用いられる Software については詳細を後述する。

4.9 NFS による user data の local 管理

4.9.1 概略

光電波では、CAD tool の利用や他の user からの file 複写、さらには高速な CPU の簡便な利用等を考えて NFS Server への user の login を許可している。rs01, rs04 等に各々の user directory (\$HOME) を個々に設けると単純に file 容量が 2 倍 3 倍と膨れ上がり、かつ似たような file がそこそこに散在するので管理の手間は大変である。このため光電波では rs?? と rs01, rs04 ではどこに login しても自らの \$HOME は唯一つしか存在しないよう考慮設定されている。

atr-rd は研究員が login する必要がほとんど皆無であること加えて、security の問題から NIS や NFS は張られていない。

rtitan01, atr-dv は全研究員が account をもつわけではないことから、\$HOME は共通化がなされていないが、rtitan01 と atr-dv 間では NFS が張られ、また、rs?? から rtitan01, atr-dv への NFS も張られている。このため、みかけ上研究員の作業 directory は 2 つ程度に絞られると考えてよい。これにより共用機の数分だけの (5 つ以上) の \$HOME directory をひとつひとつ管理する必要がなくなる。

4.9.2 NFS 管理における file 配置の手法と優劣

rs?? と rs01, rs04 で NFS/NIS による user の一括管理を行なう際は以下の 2 つの方法が实际的である。

NFS による Server-Client system の disk 構成の手法

(1) NFS Client rs?? は /etc と \$HOME を自分の disk に置く。

- 利点

(1-a) system の version up は server 側でほぼ一括してできる。

(1-b) 個人 user が自分の file を手元に置けるので back up は自分の進捗に合わせて行える。

(1-c) 小 disk 容量の場合でも結構使い度がある。通常 \$HOME が 200MB を超える user は少ない。

(1-d) 個々の \$HOME の増え方に応じて自分で disk を買い足したり取り替えできるので受益者負担が明確である。

- 欠点

(1-e) disk crash した場合、誰も助けようがなく自分で back up をしておかないと disk 内容は救済されない。

(2) NFS Client rs?? は /etc, /usr/bin, /usr/lib, /usr/ucb 等を自分の disk に置く。

- 利点

(2-a) OS のコマンド群が自前の disk にあるのでそれらを多く使用する場合は反応が速い。

(2-b) back up が root や support によって行われるので個人で back up を行う必要はない。

(2-c) 自前の disk は OS の permanent な file のみであり小容量 disk で済む。

(2-d) どの rs?? から login しても \$HOME は確保され、どの rs?? も自分の machine 同様に扱える。

- 欠点

(2-f) system の version up は rs?? の一つ一つに対して行わなくてはならない。特に Sun OS のように Patch が広く公開され豊富に出回っている場合は高い頻度で Patch を当てなくてはならないがそれが事実上不可能である。

4.9.3 光電波の NFS 管理における file 配置

光電波では前節の (1-a) 及び (1-d) のメリットが大きい。(2-a) に記した利点はほとんどの user が、more よりは less、vi よりは NEmacs というように /usr/local/bin/ の file をよく使い、また /usr/openwin/ は非常に少数であるため光電波では有効ではない。よって (1) の方式を採用している。

特に standalone machine のみではあまり必要性を感じない patch も、network 上に数十台の machine を抱える system では必要不可欠である。Patch 情報が豊富で多く当てることにより単純に信頼性が向上すると考えてよい SunOS では patch を当てないような system 管理は考えられない。このため一括して patch を当てられる (1-a) のメリットは system 管理上捨てることはできない。

4.9.4 automount の利用

4.9.2(1) の実現には automount が用いられる。自らの \$HOME を持つ rs?? に login したときは OS や application を有する server の disk を NFS mount するだけだが、Server rs01, rs04 等に login する場合、rs?? の \$HOME を Server に mount させなくてはならない。通常の NFS では全ての rs?? が常時生きていなければこのこと不可能であり、rs?? へ access がかった時のみ mount する automount が有効である。[10] [11]

光電波では user disk の mount に indirect map の automount を使用している。rs01/04 に対して rs?? に \$HOME を有する user が login した場合の automount の動作を以下に順を追って説明する。

(a) /etc/passwd に記述の /usr2/home?? に /bin/login process が access。

(b) /usr2/home?? は /auto/home?? に ln -s されており /auto/home?? が access される。

(c) /etc/auto.master の記述に指定したことにより、/auto directory は automount の管理下にある。ここへの access は /etc/auto.home の記述に従って rs01,rs04 の automount が実行される。

```
# /etc/mount rs??:/usr2/home?? /tmp_mnt/usr2/home??  
# /bin/ln -s /tmp_mnt/usr2/home?? /auto/home??
```

(d) しかも /auto/home?? は既に /usr2 から ln -s されている。

(e) このため user が /usr2/home?? に access すると一旦 /auto/home?? を経由し、 /tmp_mnt/usr2/home?? に NFS mount された rs??:/usr2/home?? が access され、 rs01, rs04 から rs?? で使用しているのと全く同一の directory が access される。

4.9.5 TITAN, CONVEX の rs?? からの automount

rtitan01 や atr-dv へも rs?? や rs01,rs04 が access できるようになっている。

(a) /usr2/convex/home ないし /usr2/titan/home に user が access

(b) /etc/auto.direct の記述に従い atr-dv:/usr2/home を mount する。

```
# /etc/mount atr-dr:/usr2/home /tmp_mnt/usr2/convex/home # /etc/mount rtitan01:/usr2/home /tmp_mnt/usr2/titan/home
```

(c) 上の indirect map に従う automount と異なり、これは直接見たい directory に mount する。

```
# /bin/ln -s /tmp_mnt/usr2/convex/home /usr2/convex/home # /bin/ln -s /tmp_mnt/usr2/titan/home /usr2/titan/home
```

みかけ上 /usr2 はどうしても rs01, rs04 に存在する directory にみえるが、これは個々の rs?? に存在しており atr-dr, rtitan01 も個々の rs?? が独自に mount を行っていることに注目すべきである。これにより直接 rs?? が rtitan01, atr-dv の file を加工することが出来、Server には負担がかからずすむ。これらは NFS mount が必要時にのみ張られるので、NFS のために CPU が不要に使われることがない、また rs?? は正しく使えば shutdown を随時行なえる等のメリットを生む。しかし使用法を正しく理解していないで local machine の shutdown 等を行なうと、全 user に対して迷惑をかける事態を引き起こすので要注意であり、研究員への正しい指導が不可欠である。

4.9.6 NFS Clinet rs?? と rs?? 同士の NFS

光電波では各 rs?? の /auto と /usr2 の間には home?? の symbolic link を行なっていない。このため client 同士間での login ではたとえ \$HOME をもつ rs?? からの login であっても NO HOME warning が返ってきてしまう。この解決法は各 rs?? の /auto と /usr2 の間に全ての rs?? の home?? に対して対応する home?? directory に symbolic link(/bin/ln -s) を張ることである。しかし常に新しい machine が購入、導入され、随時 configuration の変化する rs?? series では各 rs?? の root が新しい machine の導入毎に相当する home?? に /bin/ln -s をかけてやらねばならず、その作業量が膨大なこと、また自身の rs?? も新しい rsXX へ \$HOME を含む home?? を明示的に export(root 特権での /etc/export file の edit および root 特権での /etc/exportfs の実行) してやらなくてはならず、この作業も同じく膨大である。このため現在は rs?? 同士の login は事実上 No Home Warning を招く状態である。

4.9.7 NFS Client rs?? と rs?? 同士の login の対策の現状

いくつかの NFS Client の rs?? には個々の hostid に対応して node lock license が動作する CAD を有するので同一の研究グループ員等からの login の要求がある。これに対しては /auto との ln -s ではなく、NIS map に対して /etc/passwd に local 設定を加えて user の login directory をその rs?? のみ変更させる手法や、/usr2 に home?? の他に homeXX を実体のみでこしらえて対処する手法をとっている。これは下記に述べる automount の弊害対策として使用しているものである。

4.9.8 automount によって引き起こされる障害

automount は便利で有用であるが、mount 先の machine rs?? が生きている (= 正常動作している) か否かを一切考慮せず、指定 directory への access があると自動的に mount を試みることに大きな問題を内在させている。rs01,rs04 に対して \$HOME を有する machine が down したままであるにも係わらず annex 等から login を試みると rs01,rs04 の nfs daemon はひたすら rs??:/usr2/home?? の mount を試み続ける。このとき、login process 自体が終了しても nfs daemon 自体はひたすら mount を試みるので rs01, rs04 の負荷は大幅に上がり、たった一回の access の誤ちで server 機能が実質 down したり、nfs を新たに張れなくなってしまい、一切の login を受け付けなくなってしまう事態をひき起こす。

このような事態は rs?? が生きている時に access して rs01, rs04 から nfs が張られた後、そのことを忘却して rs01, rs04 上の process 例えば a.out や shell が rs??:/usr2/home?? を必要としているのに rs?? の shutdown を行なっても同様に引き起こされる。このような忘却は pstwm のような multi page window manager を使用している場合や kterm の終了を exit コマンドでなく window manager の pull down menu で kill する等、誤った使用法の user に数多くみうけられる。この対策としては、Xwindow の仕組みの根元からの理解の徹底と、logout で.logout などの設定をこらすことにより回避するなどが考えられる。

4.10 光電波で用いられる通常の数値計算、出力環境

本節では光電波で用いられるオーソドックスで大型計算機的な用法に際して用いられる software を述べる。

(a) Shell tcsh 6.04

光電波で事実上標準 Shell となっている。emacs や vi like な user interface を有し、使い勝手は /bin/sh, /bin/csh の比ではない。

(b) Editor

(b-1) NEmacs 3.3.2

多くの本で解説されているので一般的な解説は除くが fortran.el を用いることにより fortran の indent 等に対応したり C-mode.el を用いることにより C 言語に対応できる。尚、これらの拡張は edit する file の description を自動的に見分けて自動的に導入される。

(b-2) vi

jstevie を用いることにより日本語入力が可能であり、kterm 5.2.0 と kinput2 の併用で inline 入力となる。

(c) C compiler

(c-1) /bin/cc

SunOS 4.1.3 に添付の cc。-cg92 を support していないため ss10 用に特化した命令による高速化は出来ないが、それでもいくつかの事例で /usr/lang/acc との差が大きいことが明らかである。いわゆる K&R 版 C の文法とその関数に基づく。OS のライセンスさえあれば利用可。

(c-2) /usr/local/bin/gcc

Free Software Foundation の GNU project 製の cc。高速でよく吟味されており debugging も順調で早いペースで version up する。gcc 2.5.x からは -mv8 option も用意され SPARC version8 から導入された命令セットを用いた整数の掛け算割り算の高速化も活用出来る。option 無しだと ANSI-C だが -traditional および -fixinclude option で K&R-C にも対応し、debugger として同じく GNU project の gdb が用意されている。

(c-3) /usr/lang/acc

SunSoft 製 ANSI-CC。有料で OS と別売である。flexlm ライセンスマネージャの管理下にあり、光電波では同時に 1 本しか使えない。速度的には gcc と同等かそれ以下という事例が多い。-cg92 option を有し SPARC version8 から導入された floating point 演算命令セットを活用出来る。-Xs option で /bin/cc と同等の動作をする。debugger は dbxtool か dbx だが SunWorks の dbx は openwindow 専用なので光電波の標準環境と混在で使用できない。

(d) Fortran77 compiler

(d-1) /usr/larg/f77-1.4

旧版 f77。option -cg92 を有せず、SPARC version8 から導入された floating point code は support しない。しかし現在光電波で最も利用される数値演算ライブラリ IMSL 及びグラフ作成ソフト dissp11 はこの f77-1.4 でしか動作しない。よって存在は不可欠である。生成する code も /usr/larg/f77 に比べて実行速度でほとんど変わらないか速い場合さえもある。

(d-2) /usr/local/bin/f2c

AT&T Bell Lab. と Bellcore が作成した Free Software。f77 の code を C-language の code に変換する。このため hogehoge.f という file を f2c に通すと hogehoge.c という名称の file が作成される。f2c というコマンドだけでは毎回 f2c 専用の library の link 指定をしなくては行けないので、fgcc 及び fcc という fortran compiler command like に使える shell script が用意されている。

(d-3) /usr/lang/f77

新版 f77-2.01。-cg92 option を有する。しかし対応する数値演算ライブラリや Graphic tool がないこと、flexlm liscence maneger により光電波では同時に 3 本しか使えない等制約がある。速度的にも f77-1.4 に比して抜群に速いというわけではない。

(e) 数値演算ライブラリ

(e-1) IMSL

世界的に知られた Fortarn77 用数値演算ライブラリ。大きく 3 つに分かれ SFUN, STAT, MATH から成る。flexlm により同時に 2 本しか使えない。かつては source code 提供サービスもあったが現在は binary のみで、コンパイルが -cg87 option でなされているので三角関数や一部の演算では実行速度が遅くなる。

(e-2) C/MATH

IMSL の C-language 用数値演算ライブラリ。flexlm により同時に 1 本しか使えない。

(e-3) SSL-II

世界的に知られた Fortarn77 用数値演算ライブラリ。IBM machine ではほぼ標準の数値計算ライブラリ扱いであった。SSL は source code も公開されていたが本ライブラリは binary 供給である。光電波では 1CPU 分のライセンスを有する。

(e-4) NETLIB

usenet 等 Internet 上に流布される Free Software 群。source code 提供であり自由に compile できるので CPU のバージョンに応じて最適な compile が出来、実行速度が速い。但し、いくつかのサブルーチンは buggy である。netlib では fftpack, vfftpack, eispack 等はよくこなれており評価は高い。

(f) グラフ作成ツール

(f-1) disspla

光電波での現在の版は version 11。rs01, rs04 上に各々一本計二本の licence を有する node lock 形購入を行なっている。Fortran 77 の subroutine の形で提供され、user の Fortran program に組み込む形で動作させる。対応 Fortran compiler は f77-1.4。使い勝手は悪く GUI とは程遠いが細かいところまで全て手が届き各種プリンター形式以外にも eps 形式も光電波作成の diss2eps を用いることにより可能なので本格的な図面作成と \LaTeX でのグラフ出力に不可欠である。

(f-2) gnuplot

ダートマス大学で作成された Free Software。GNU project の製品ではない。interactive に使用する pvwave-CL 的な使い勝手で、式を与えると数値曲線を描画することを基本としている。加えて外部 file の ascii data も awk 的なセンスで読み込める。出力 file の format が多彩でかつ command 入力も emacs 的な key assignment を有し、親和性と操作性にすぐれているため簡便な 2/3 次元グラフの作成に光電波では広く用いられている。

(f-3) xgraph

gnuplot と同様に簡便に使用できる UC berkeley 校で作成された描画ツール。Free Software。外部 file の ascii data を awk 的なセンスで読み込める。出力 file には HPGL, postscript, InterViews-3.1 idraw が選択できる。GUI が gnuplot より良好なので利用者も多い。

(f-4) Spyglass

Contour 図面や 3 次元 plot 等に有効な spyglass 社の製品。Motif 風の GUI を有する。入力は ascii file でも数式でも可。現在 flexlm の管理下で光電波では一本を有する。電磁界分布の図示等に有効。

4.11 図形出力の file 形式

実際に計算が終了した後、その数値をそのまま結果として利用することもあるが、通常はグラフ化して print out することが多い。また \LaTeX 等に取り込んだり OHP 用に手を加えることも多い。このような場合 Fortran 77, C 等の結果を一旦 ascii で file に落とし、次にグラフ化して図形データに変更し、さらにその図形データを次の形に変更するというような作業が伴う。このとき、どの形式の file にするかにより後々の加工が困難になったり自由度が大きくなったりするので、よく考えて file 形式を選択しなくてはならない。ここではグラフ作成ツールが出力する形式や printer へ入力する形式として知っておかなくてはならない file 形式について記述する。

(g-1) plain text(ascii)

日本語を伴う場合、JIS, EUC, SJIC と呼称される文字コードを含むが、通常 Fortran 77 は multibyte 文字をうまく処理できないので計算出力は ascii である。文字コードについては [12] に詳しい。kterm 使用時は上記 4 つの code 系は全て表示でき NEmacs や Mule でも処理できる。plain text をそのまま出力できる printer は LN05(ln) であり、QMS860(qms) も出力可能だがあまり使用されない。ln は printcap で nkf を処理途中にかましているのて上記 4 つの code 系を全て正しく印刷できる。

(g-2) tektronix 401x

tektronix 4010 terminal 上で描画する際に用いられる形式。中身は ESC code や control code を大量に含んだ ascii code である。kterm や xterm では、そのまま cat すれば preview でき、かつ LN05(ln) で印刷できる。この形式はあとで絵や文字に手を加えることが困難なこと、font が選べないことなど不便をかこつため最近 user が少ない。しかし postscript に比して error が少ない。膨大な点数を有する plot では file が小さい等利点もあり、特に dispsla user に利用者が残存する。白黒のみ。

(g-3) postscript

printer として QMS860(qms), PC-PR602ps(pc), Apple Laser Writer NTXj II (lw), tektronix PHASER II (tek) の 3 種が使い、光電波で最もよく利用される形式。基本的に ascii code で日本語部分は JIS,EUC,SJIS code を使用している。このため editor で直接 file に手を入れることができ、sed 等で余計な命令を省くことも可能である。

一旦 postscript にすると ghostscript, ghostview での preview が可能な他、ghostscript で tek4010, HPGL, LIPS III, LIPS II, PCL II, PCL III 等各種 printer の形式に合わせて加工できるため自由度は高い。現在ではさらに自由度の高い eps を利用する user も多い。postscript に「準拠」と称する互換 printer が多く出ているが光電波のように高度な postscript file を出力する user の要求に耐えられる互換機は多くないようである。純正 adobe postscript interpreter を有する printer は光電波においては不可欠である。カラーにも対応する。

(g-4) encapslated postscript (eps)

postscript 形式を拡張してさらに自由度を上げ、大きさや縦横比を自由に変える等が可能な形式。 \LaTeX user

には不可欠な形式である eps は、光電波で有する多くの Free Software 群が出力可能であり、むしろ市販の Software でこれに対応する tool は些少である。disspla は光電波で開発された diss2eps という tool を使用することにより eps に変換可能である。postscript printer で出力可能。

(g-5) idraw

idraw は InterViews project の InterViews-3.1 idraw という描画 tool の通称である。ここで使用される file の形式は eps の拡張形である。gnuplot, xgraph 等がこの形式を support しており、出力されたグラフをそのまま細かい点まで加工して後に postscript printer に直接出力できるので user は多い。もちろん \LaTeX でとり込める。tool としての idraw は若干 buggy で postscript error を引き起こしたり、file 形式ゆえに部品数が多くなると一気に遅くなったり、同じ idraw でも InterViews 2.6 版と InterViews 3.1 版で互換性が一部無かったりして不便をかこつこともあるがその便利さ上記不便さをしのぎ、また user interface が Macintosh 上の MacDraw II とよく似ているため取りつきやすく、user は bug をうまく避けながら愛用している。特に text object に関して bug が発生しやすいことが経験から明らかである。

(g-6) HPGL

HP の plotter を利用する場合に広く利用される。但し光電波 LAN に直接接続された HPGL plotter, printer は無く、もっぱら実験室内で使用されるにとどまっている。光電波の Free Software 群では HPGL の file を出力するものが多く、また ghostscript も postscript を HPGL に変換できる。

(g-7) xwd

Xwindow の bitmap image を dump した形式。いくつかのアプリケーションではどうしても「画面に出るが printer に持っていけない」場合に遭遇するのでこのときは強制的に /usr/X11R5/bin/xwd で画面を切り取り xwd 形式に落とし、しかるのちに xpr で postscript や tektronix 401x 形式に変換することが必要となる。光電波では /usr/X11R5/bin/xgrabsc という Free Software を用いて xwd を経ることなく直接 postscript に変換しているのであまり用いられない。[13]

(g-8) dvi

「DeVice Independent」の略。文字コードを含む binary code。 \TeX のコンパイルが終了した後に生成されるフォーマット。xdvi で preview することにより印刷前に印刷イメージを画面で check できる。一旦 dvi 形式にすると、postscript や LIPS II, LIPS III 等に変換できるという利点は大きい。光電波ではこの dvi file を直接 edit する tool はなく単に中間 file としてのみ使用している。xdvi という previewer は操作性も表示も秀逸であり dvi file に埋めこまれた eps もきれいに表示する。

(g-9) xfig

X11R5/contrib/xfig が使用する eps の拡張形式と思われる形式。ascii code。xfig は transfig と呼ばれる tool で postscript に変換できる。xfig の最大の特徴は図形を入力するとそれに対応する \LaTeX のスクリプトを pic-

ture 環境の形で出力できることで（光電波の tool は未だ整備不良で細かい object や文字などは変換できない）
fig2dev, transfig 等の Free Software tools が揃っている。

(g-10) tgif 用 obj 形式

tgif で用いられる図形 file。binary code。gnuplot が出力形式として support している。tgif はこの code を
使用しているが idraw 形式に比べコンパクトなので処理速度は速い。eps 等各種形式に変換できる。

5 購入時の事例

光電波の unix 計算機環境での購入時のベンダーとのトラブルについて述べる。Macintosh, IBM-PC と比して unix workstation は管理が困難といわれているが、それは単に運用者の心掛けの一つであり、十分な user の協力があれば十分な使い勝手をもつ環境を提供できる。また、業務上の書類を扱う上でも Macintosh や Windows 3.1 on MS-DOS 5.0(or higher) に比して unix WS は安定しており、(年単位でしか fail しない) 研究用に適していると考えられる。管理にあたっては導入する Software や Hardware の情報を良く把握してはならない。常に最速の機種や Software が発表されるが、それが即座に光電波に良いとは限らずむしろ逆であるケースさえしばしば露見する。

(a) 単に Fortran 77 や C の計算速度を求めている場合は個人用 WS のみではなく共用機にも解を求める方が正しい。実際共用機とほとんど同じ計算速度を示す数字 MFLOPS, SPECfp'92, SPECint'92 等が出されているが個人用計算機では disk I/O が多い計算の場合 SCSI が bottle neck となる可能性が高い。最近 RAID-7 等 SCSI でも disk stripping を施して disk I/O を高速化する試みがなされているとはいえ、やはり disk interface が専用 I/O bus で直結されているものの方が格段に速いため I/O を伴う計算では大きな差がつく。全て RAM 上で計算するような program だけでは不可能な場合として以下に例を示す。

- 途中経過を常に file に書き出す方式を取っている program。
- IBM, VAX 等その program がかつて使用された machine の condition に依存して計算領域の大きさを絞り外部 file を用いて解決している program。
- 仮想記憶を support していない UNIX V.32 時代や PDP-11 等の Mini computer 等で使われていた subroutine を使用している場合。

(b) X11 の表示速度を著しく向上させるような Graphic Card はそれ専用の Xserver 用 binary が必要である場合が少なくない。光電波で採用している GUI は常に最新でかつ安定している X11 を用いているので (現在は X11R5 patch level 25) メーカーより 1 年～1 年半は新しい Window system を使用している。メーカーの最新の Graphic board がこれに対応するまで時間がかかる。また、製品によっては発売時に一番流通しているリリースに対してのみ server を binary で提供し board 自体が price list から外れるとその後一切 Support されないことも多い。

- Sun Sbus GS Graphic board 用 Xserver binary である "Xsun24" は X11R4 では入手可能だったが X11R5 では正式 support は一切されない。また GT については情報さえも容易には入手できない。
- Sun Sbus SX Graphic board は X11R5 の Xserver が動作するが、その X11R5 の server binary は SunOS 5.x 以上でないと動作しないので SunOS 4.1.x 上では高速性が保証されないだけでなく動作自体も保証されない。

(c) 新しい OS は常にそのメーカーの最新機種の高性能のものに合わせて作られるので、光電波のように performance が 12MIPs 近辺から 100MIPs 以上の WS が各世代に渡り使われている site では古めの OS の方が全体の益である。

- SunOS 5.0 では OS が重た過ぎて、SS1 級の machine では OS version up による利点より処理速度の低下のデメリットが決定的に大きいとのアドバイスを vender 側からもらった。
- SunOS 5.3 でも「2CPU 以上を有する Machine の性能向上」を基本に開発が進むため、Kernel 部分に手を加え続けており、SunOS 5.2 の bug fix version というわけではなく、必然的に並行処理に関わる部分の bug 混入が激しくなる。

(d) 営業の持ってくる話と技術担当者の持ってくる話が違う。どうしても営業は売ること中心で積極的であり、技術は support の責任があるため保守的である。このため、新製品 (や旧製品についても) 常に両方から情報を提示してもらう。Internet ではかなり信憑性の高い情報が流れているので新しい製品の情報の入手には comp.sys.sun.admin, comp.sys.sun.hardware 等の講読は欠かせない。

(e) Sofoware の購入について。通常光電波における CAD は新しい研究の方向に合わせて柔軟に導入が行なわれるため得てして unix の知識も浅くまた光電波の環境にも知識のない研究員が購入担当者になるケースが多い。このため必要な注意事項を check しないまま購入に及んだり不必要な Software を押しつけられてしまう case が考えられる。

実際に生じた事故例から考えると、通常経験が少々あっても「光電波が sun を使用している」という言葉だけからでは CAD Software の営業は以下のように細部を取り違えて補完すると考えるのが無難と思われる。

項目	取り違い	光電波の実情
(1) OS	SunOS 5.3	SunOS 4.1.3
(2) 日本語	日本 Sun 製の JFP	Wnn. Canna. kinput2
(3) keyboard	日本語 Type V	TypeIV unix
(4) network	stand alone	Client Server
(5) printer	NeWS print	BSD lpdaemon
(6) license manager	存在しない	flexlm
(7) X11server	Open Window	MIT sample server
(8) Window manager	Open Look	pstwm, twm

このように悉く異なるため、あとあと大きなトラブルを招く。また逆に「Motif が必要」ということで光電波への導入を断念した Software でもよく確認したところ Xlib のみ使用して実には支障は無い場合であること

が多い。逆に、Motifは version や vender によって動作が違うので「Motifは Motifでも IXI Motif 1.2 ではないとダメ」というケースもよく発生しているようである。

- (f) 購入研究員と root 担当研究員の協力。server root を担当する研究員は「相談を持ち掛けられたら相談にのる」という対応でないと計算機に関わる時間が多過ぎて自分の研究がおろそかになる。しかし、得てして計算機購入担当者は server root が計算機購入の全てを最終段階で check して、server root 側で足りない部分を補ってくれるものだと誤解しがちなので、実際物品が納入されてから購入伝票の不備が見つかったりケーブル等必要な部品の発注手違いで光電波の network に接続出来ないことが明らかになるケースがある。server root は光電波の computer network 全体を把握しているわけではないので、購入担当者が事前に server root と入念に打ち合わせる。できれば伝票まで相互に check することにより誤解を生じないよう連絡関係を取る必要がある。また、余剰機器の下取り等によりさらなる値引きが期待出来る時もあり研究費の節約のためにも絶えず製品情報と直接接している server root との連絡が欠かせないであろう。

6 支援作業

Server-Client system を中心とする computer network は一旦組み上がってしまえば上述のように複雑な機能を高度に使いこなす需要に答えるものである。しかし注意しなくていけないのは、そのような安定した system も外部の動向によっては一夜にして改変を行ない、新しい system に切り変わらなければ組織全体の足を引っ張ることとなる。このため LAN で相互接続された computer network system は組み上がって完動していても常に自分に接続されている network の情報を入手しその将来計画を鑑みて全体を組み換える準備を進めなくてはならない。「network 生き物である」といわれる由縁である。

このため、ある程度以上の skill を有する人間を常に管理者として要求する。研究員の root 作業および予算配分や新機種新ソフトウェア情報の収集等に関する負荷を下げるためには外部の計算機管理 / 支援作業者に依託することが多く行なわれる。

しかし現在、このような computer network を管理する人間の skill を正しく客観的に評価する基準が確立していない。computer network の管理の外部依託は組織の担当者の主観的眼識によってその担当者を評価することに頼っている場合が大半である。

このため光電波では客観的にかつ漏れの少ない評価を行なうため、外部依託する場合には管理者 / 支援作業者の skill を判定する skill check sheet を作成している。もちろんこの sheet だけでは不完全であり、最終的には面接を行なう必要があるが、その面接の前にその候補者の skill を有る程度判定出来、面接をスムーズに行なえるだけでなく、依託先に光電波の要求する技術者の skill を伝えることにもなり有効であると考え。次頁からその check sheet を掲載する。

システム管理者としての経験値の水準に関する質問

以下の質問に○×でお答え下さい。なお、「したことがある」等の表記はあなたご自身が主体的、指導的立場で実施されたか否かを問うておりますので、どなたか他の方の支援として実施なされた場合は×でお答え願います。質問の意味が不明とお感じの場合は△印をお付け下さい。

以下、点数1。skillful な user level

- 新規 user の追加と、最低限必要な rc files を滞り無く準備出来る。但し、 /etc/cshrc や /etc/profile 等に頼らずゼロから書き上げられること。 [14]
- OS の再 configuration[15] 出来る。その際、 configuration で指定する各 parameter が組み上がった kernel 内部でどのような働きをするかの概略を示せる。
- VT terminal, SCSI devices, RS232C 等の最低限の hardware の知識があり、 RS232C の null modem cable 配線を自作出来る。 [16]
- X11R5 の使用経験があり、 MIT sample ないし Motif Window manager の設定を自分で書ける。
- BSD と system-V の system 管理上の相違点はだいたい頭に入っていて5つ以上そらで列挙することが出来る。
- ANSI-C と K&R-C の相違点はだいたい頭に入っていて3つ以上そらで列挙することが出来る。
- Fortran77 の VAX/VMS extention の部分はだいたい頭に入っている。
- NEmacs, Mule, MH, GNUS について知識があり、 .emacs を触れる。
- MH6.8 以上の .mh_profile, .mhl.format, replcomps, forwcomps を自分で調整したことがある。
- ed,sed,awk,tr,tee,grep,sort,uniq の基本使用法はマニュアルを必要としない。 [17] [18]
- 10Base5 Yellow Cable に治具を用いて穿孔し、 MAU ないし Tranceiver を取りつけた経験がある。
- jL^AT_EX の使用経験があり、簡単な style file の改造なら出来る。
- xmkmf についての基礎知識を有し、使用経験がある。 [19]

以下、点数3。一般的な root level

- NIS の Yellow Page Service 管理下の file を4つ以上そらで列挙出来る。
- lpshed をきちんと設定出来、 remote print を設定した経験を有する。
- X11R4 ないし X11R5 の make world の経験を持つ。
- Postscript file をいじって使用 font を変えたことがある。
- NIS の設定は経験があり、 user の追加程度ならばマニュアルは不要である。

- etherfind,netstat,vmstat,iostat,nfsstat 等を使用して system performance tuning を行なった経験がある。

[20]

- inetd,routed,gated を設定出来る。
- archie ないし xarchie で外部の FTP site へ sources を取りにいける。
- ed のみを用いて自由自在に ascii code file を書き換えられる。
- perl script が読み書き出来る。

以下、点数 5。光電波における従来の root level

- gcc2.3.3 以上を make し、完動を check した経験を有する。
- X11R4 ないし X11R5 の contrib package を含む make world の経験を持つ。
- Emacs の make の経験を持ち、大概の elisp file ならば manual を見ながら書き換えられる。または site.el を書いた経験がある。
- Wnn ないし Canna の make および個人辞書の設定経験をもつ。
- INN ないし CNEWS の設定が出来、NNTP についての知識がある。
- CAP 6.0 以上を install、設定出来る。lwsrv 等の daemon を設定出来る。
- jL^AT_EX の install 経験があり、dvi2ps における必要な font を 2 種類以上そらで列挙出来る。
- sendmail-MX と DNS を用いない sendmail の設定項目の違いがだいたい頭に入っている。
- DNS の設定と daemon の起動。さらに DNS の動作の check を行なえる。
- sendmail.cf の設定がマニュアルを読みながらなら出来る。
- uucp と modem の設定、接続が出来る。
- flex lisence maneger を使用して license 管理を行なった経験がある。

7 まとめ

以上、光電波における計算機運用の実際につき、トラブルレポートを含めて述べてきた。表向きの体裁の良い運用は多くの本で出ているので参考として欲しい。メーカーの欠点や営業の対応の悪い点は印刷物になりにくいので、せいぜい雑誌 (unix magazine 等) でほんの少し現れる程度だが実際はメーカーの把握している bug 情報や営業の知識は user とほとんどかわらないということを前提に行動したほうが良いことが圧倒的に多い。これらの情報についての他の窓口としては Internet を活用することで得られる。きれいな色刷りのパンフレットや高価な見積書の金額につられて全て市販の Software でシステムをかためるのはよくあるケースである。しかし、多くの場合、package software vender は stand alone の WorkStation に自社の software のみを (場合によっては自分の事業所の software のみを) 載せて動作させることを前提としておりいくつもの software を同一の WorkStation 上で動作させることに対して否定的で時には動作保証を与えない。Free Software は、user を通して同一の WorkStation 上で動作させられるように常に改良版を提供してくれるのでうまく使いこなすことで、見栄えだけは良くないが安価かつ高度な tool を単一の環境で組み立てることを可能にする。

また、全てを Free Software だけで頼るのも困難な場合があり、現在の計算機の動向を鑑みると Motif 等の導入の真剣な検討等が必要な時期となっている。また全てのメーカー製 Software を否定しては拡張は不可能である。このようなことも念頭において、今後の光電波の運用にこのレポートが生かされれば幸いである。

8 謝辞

本レポートの作成と本レポート記載の計算機運用を進めるにあたり御指導、御討論頂いた(株)ATR 光電波通信研究所 1993 年度計算機委員である、下田平研究員(計算機委員長)ならびに大田原研究員に感謝致します。

9 Appendix-I

GNU EMACS GENERAL PUBLIC LICENSE

(Clarified 11 Feb 1988)

Copyright (C) 1985, 1987, 1988 Richard M. Stallman Everyone is permitted to copy and distribute verbatim copies of this license, but changing it is not allowed. You can also use this wording to make the terms for other programs.

The license agreements of most software companies keep you at the mercy of those companies. By contrast, our general public license is intended to give everyone the right to share GNU Emacs. To make sure that you get the rights we want you to have, we need to make restrictions that forbid anyone to deny you these rights or to ask you to surrender the rights. Hence this license agreement.

Specifically, we want to make sure that you have the right to give away copies of Emacs, that you receive source code or else can get it if you want it, that you can change Emacs or use pieces of it in new free programs, and that you know you can do these things.

To make sure that everyone has such rights, we have to forbid you to deprive anyone else of these rights. For example, if you distribute copies of Emacs, you must give the recipients all the rights that you have. You must make sure that they, too, receive or can get the source code. And you must tell them their rights.

Also, for our own protection, we must make certain that everyone finds out that there is no warranty for GNU Emacs. If Emacs is modified by someone else and passed on, we want its recipients to know that what they have is not what we distributed, so that any problems introduced by others will not reflect on our reputation. Therefore we (Richard Stallman and the Free Software Foundation, Inc.) make the following terms which say what you must do to be allowed to distribute or change GNU Emacs.

COPYING POLICIES

1. You may copy and distribute verbatim copies of GNU Emacs source code as you receive it, in any medium, provided that you conspicuously and appropriately publish on each copy a valid copyright notice "Copyright (C) 1988 Free Software Foundation, Inc." (or with whatever year is appropriate); keep intact the notices on all files that refer to this License Agreement and to the absence of any warranty; and give any other recipients of the GNU Emacs program a copy of this License Agreement along with the program. You may charge a distribution fee for the physical act of transferring a copy.
2. You may modify your copy or copies of GNU Emacs source code or any portion of it, and copy and distribute such modifications under the terms of Paragraph 1 above, provided that you also do the following:
 - a) cause the modified files to carry prominent notices stating that you changed the files and the date of any change; and

- b) cause the whole of any work that you distribute or publish, that in whole or in part contains or is a derivative of GNU Emacs or any part thereof, to be licensed at no charge to all third parties on terms identical to those contained in this License Agreement (except that you may choose to grant more extensive warranty protection to some or all third parties, at your option).
- c) if the modified program serves as a text editor, cause it when started running in the simplest and usual way, to print an announcement including a valid copyright notice "Copyright (C) 1988 Free Software Foundation, Inc." (or with the year that is appropriate), saying that there is no warranty (or else, saying that you provide a warranty) and that users may redistribute the program under these conditions, and telling the user how to view a copy of this License Agreement.
- d) You may charge a distribution fee for the physical act of transferring a copy, and you may at your option offer warranty protection in exchange for a fee.

Mere aggregation of another unrelated program with this program (or its derivative) on a volume of a storage or distribution medium does not bring the other program under the scope of these terms.

3. You may copy and distribute GNU Emacs (or a portion or derivative of it, under Paragraph 2) in object code or executable form under the terms of Paragraphs 1 and 2 above provided that you also do one of the following:

- a) accompany it with the complete corresponding machine-readable source code, which must be distributed under the terms of Paragraphs 1 and 2 above; or,
- b) accompany it with a written offer, valid for at least three years, to give any third party free (except for a nominal shipping charge) a complete machine-readable copy of the corresponding source code, to be distributed under the terms of Paragraphs 1 and 2 above; or,
- c) accompany it with the information you received as to where the corresponding source code may be obtained. (This alternative is allowed only for noncommercial distribution and only if you received the program in object code or executable form alone.)

For an executable file, complete source code means all the source code for all modules it contains; but, as a special exception, it need not include source code for modules which are standard libraries that accompany the operating system on which the executable file runs.

4. You may not copy, sublicense, distribute or transfer GNU Emacs except as expressly provided under this License Agreement. Any attempt otherwise to copy, sublicense, distribute or transfer GNU Emacs is void and your rights to use GNU Emacs under this License agreement shall be automatically terminated. However, parties who have received computer software programs from you with this License Agreement will not have their licenses terminated so long as such parties remain in full compliance.

5. If you wish to incorporate parts of GNU Emacs into other free programs whose distribution conditions are different, write to the Free Software Foundation. We have not yet worked out a simple rule that can be stated here, but we will often permit this. We will be guided by the two goals of preserving the free status of all derivatives of our free software and of promoting the sharing and reuse of software.

Your comments and suggestions about our licensing policies and our software are welcome! Please contact the Free Software Foundation, Inc., 675 Mass Ave, Cambridge, MA 02139, or call (617) 876-3296.

NO WARRANTY

BECAUSE GNU EMACS IS LICENSED FREE OF CHARGE, WE PROVIDE ABSOLUTELY NO WARRANTY, TO THE EXTENT PERMITTED BY APPLICABLE STATE LAW. EXCEPT WHEN OTHERWISE STATED IN WRITING, FREE SOFTWARE FOUNDATION, INC, RICHARD M. STALLMAN AND/OR OTHER PARTIES PROVIDE GNU EMACS "AS IS" WITHOUT WARRANTY OF ANY KIND, EITHER EXPRESSED OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY AND FITNESS FOR A PARTICULAR PURPOSE. THE ENTIRE RISK AS TO THE QUALITY AND PERFORMANCE OF THE PROGRAM IS WITH YOU. SHOULD THE GNU EMACS PROGRAM PROVE DEFECTIVE, YOU ASSUME THE COST OF ALL NECESSARY SERVICING, REPAIR OR CORRECTION.

IN NO EVENT UNLESS REQUIRED BY APPLICABLE LAW WILL FREE SOFTWARE FOUNDATION, INC., RICHARD M. STALLMAN, AND/OR ANY OTHER PARTY WHO MAY MODIFY AND REDISTRIBUTE GNU EMACS AS PERMITTED ABOVE, BE LIABLE TO YOU FOR DAMAGES, INCLUDING ANY LOST PROFITS, LOST MONIES, OR OTHER SPECIAL, INCIDENTAL OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OR INABILITY TO USE (INCLUDING BUT NOT LIMITED TO LOSS OF DATA OR DATA BEING RENDERED INACCURATE OR LOSSES SUSTAINED BY THIRD PARTIES OR A FAILURE OF THE PROGRAM TO OPERATE WITH PROGRAMS NOT DISTRIBUTED BY FREE SOFTWARE FOUNDATION, INC.) THE PROGRAM, EVEN IF YOU HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES, OR FOR ANY CLAIM BY ANY OTHER PARTY.

10 Appendix-II

光電波通信研究所における Sun Workstation の Hardware 個別一覧

番号	使用者氏名	S/N	機種	主記憶	二次記憶	備考/補助装置
rs01	Server	(SN 307K0012)	630MP	128MB	(2+4)MB	
rs02	atr-rd	(SN 142F0680)	ss2	64MB	(0.4+2+1)GB	
rs03	(旧 澤田)					
rs04	Server	(SN 048K0238)	670MP	128MB	(2+2+1)MB	
rs05	support	(SN 112F2704)	IPC	16MB	0.2GB	
rs06	山賀	(SN 112F2708)	IPX+	64MB	1GB	
rs07	Davis	(SN 112F2591)	IPX	64MB	1GB	
rs08	木村	(SN 407M2427)	ss10/51	64MB	1GB	
rs09	大鐘	(SN 113F5314)	ss2	32MB	0.4GB	
rs10	堀江	(SN 145F1559)	ss2	32MB	0.4GB	暫定使用
rs11	三浦	(SN 306M3846)	ss10/41	32MB	1GB	
rs12	(MMIC)	(SN 941F1144)	ss1	16MB	0.4GB	SunOS 4.1.1
rs13	千葉	(SN 042F1655)	IPC	20MB	0.2GB	
rs14	高橋	(SN 042F1667)	IPC	20MB	0.2GB	
rs15	村上	(SN 042F1701)	IPC	20MB	0.2GB	
rs16	(MMIC)	(SN 112F0675)	ss2	32MB	0.4GB	
rs17	竹内	(SN 031F1019)	IPC	24MB	0.2+ 外部 0.4GB	
rs18	三村	(SN 112F2595)	IPC	20MB	0.2GB	
rs19	電波暗室	(SN 113F3528)	ss2	32MB	0.4+ 外部 1GB	GPIB
rs20	澤田	(SN 144F3169)	ss2+	64MB	0.4GB	
rs21	大田原	(SN 214F0354)	ss2+	32MB	0.4+1GB	
rs22	Thomas	(SN 214F0365)	ss2	32MB	0.4GB	
rs23	無 2 実験室	(SN 219F0614)	ss2	64MB	0.4GB	GPIB + bus exp.
rs24	下田平	(SN 241F1631)	ss10/30	32MB	0.4GB	
rs25	土居	(SN 241F1395)	ss10/30	64MB	0.4+1GB	
rs26	山田	(SN 249F7207)	ss10/30	128MB	0.4+ 外部 1GB	
rs27	(MMIC)	(SN 318M1849)	ss10/30			CRT17"
rs28	新上		ss10/51	32MB	1GB	

参考文献

- [1] 斉藤明紀, 山口英: “Unix communication notes(36): NIS”, UN I X magazine, 6, 6, pp. 61 – 71 (1991).
- [2] H. Stern: “NFS & NIS”, NutShell handbooks, O’ Reilly & Associates, Inc., ASCII (1992).
- [3] C. Liu and P. Albits: “DNS and BIND”, NutShell handbooks, O’ Reilly & Associates, Inc. (1993).
- [4] 錦見美貴子: “たまご version 2.24 マニュアル”, 電子総合研究所 (1990).
- [5] “かな マニュアル version 2.2”, 日本電気 (1993).
- [6] 岩熊哲夫, 古川徹生: “ \LaTeX のマクロやスタイルファイルの利用 version 2.10”,
bear@hashi1.civil.tohoku.ac.jp (1993).
- [7] 伊藤和人: “ \LaTeX トータルガイド”, 秀和システムトレーディング (1992).
- [8] 奥村清彦: “ \LaTeX 美文書作成入門”, 技術評論社 (1991).
- [9] 荒井美千子: “Nemacs 入門 (15) cmutex”, UN I X magazine, 8, 1, pp. 95 – 116 (1993).
- [10] 山口英: “Unix communication notes(42): automount”, UN I X magazine, 6, 12, pp. 33 – 42 (1991).
- [11] 三膳孝通: “管理者の理想と現実 (2): automount の導入”, UN I X magazine, 7, 4, pp. 104 – 108 (1992).
- [12] 荒井美千子: “Nemacs 入門 (22)”, UN I X magazine, 9, 3, pp. 138 – 149 (1994).
- [13] 遠藤知宏: “続ウィンドウシステムについて (10): xwd フォーマット”, UN I X magazine, 8, 10, pp. 61 – 70
(1993).
- [14] 山口和紀: “The UNIX Super Text (上下)”, 技術評論社 (1992).
- [15] 岡山聖彦, 片山善章, 馬場健一: “スーパーユーザーへの道 (5)”, UN I X magazine, 6, 9, pp. 34 – 56 (1991).
- [16] 岡山聖彦, 片山善章, 馬場健一: “スーパーユーザーへの道 (1)”, UN I X magazine, 6, 5, pp. 36 – 67 (1991).
- [17] D. Dougherty: “sed & awk プログラミング”, NutShell handbooks, O’ Reilly & Associates, Inc., ASCII
(1991). 原題 sed & awk.
- [18] A. Aho, P. Weinberger and B. Kernighan: “プログラミング言語 AWK”, トッパン (1991). 原題 The AWK
Programming Language.
- [19] P. DuBois: “Software Portability with imake”, NutShell handbooks, O’ Reilly & Associates, Inc. (1994).
- [20] M. loukides: “UNIX システムチューニング”, NutShell handbooks, O’ Reilly & Associates, Inc., ASCII
(1991). 原題 System Performance Tuning.