

## ノートブックコンピュータ上での 移動 UNIX 環境の実現と評価†

杉浦 一徳‡

石井 公夫§

村井 純¶

慶應義塾大学 環境情報学部

ワークステーションによるキャンパスのコンピュータ環境と同様な環境をインテル 386 アーキテクチャのノートブックコンピュータ上に実現した。BSD UNIX と X ウィンドウを用い、TCP/IP プロトコル体系によって接続される。実用的な環境を実現するためには、狭いスクリーンで使用するためのアプリケーションの移植と改造、移動ノードへの対応、少ないディスク容量の有効利用、キャンパス管理システムとの協調、セキュリティといった諸問題を解決する必要があった。本論文ではこれらの問題を議論し、試作システムの評価を報告する。

### 1 はじめに

移動が可能なコンピュータの性能が向上し、従来ワークステーションによって提供されていたキャンパスコンピュータ環境と同等の環境がノートブックで提供できるようになった。しかし、実用に向けてはグラフィックインターフェースやメモリ空間、ディスク空間などの限定された資源の問題に加えて、移動体としての通信技術や、分散システムへの整合性などの課題を解決する必要がある。

ここでは、1990年に開校された慶應義塾大学湘南藤沢キャンパス(以後SFCとする)における学生の所有するノートブック型コンピュータを対象に、同キャンパスのUNIXワークステーション(WS)を中心として構成されるキャンパスネットワーク(CNS)[1][2]との統合的な利用を目的として開始した、ノートブック環境の研究・開発活動について報告する。

現在のCNSはFDDIを基幹としEthernetを支線とするネットワーク環境上に、各種サーバ群、サービス環境、分散ファイルシステム、管理運用システムが稼働し成立している(図1)。利用者はワークステーションから、文書作成、電子メールなどの機能を用いて日常の教育、研究、事務活動を行なう。

NFSによってマウントされた大学内の共通ホー

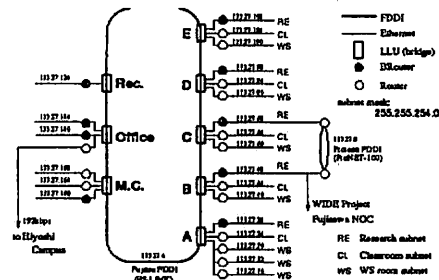


図1: 慶應義塾大学湘南藤沢キャンパスネットワーク(CNS)

ムサーバによって、学生はキャンパス内のどのWSを利用して同一の環境が得られる。アカウントは各職員・教員・学生に全て用意され、NISによって管理される(研究室内で利用される独自のWSのアカウントはこれにあてはまらない)。以後これをWS環境と呼ぶ。

学生はMSDOS用のノートブックコンピュータ(以後ノート機とする)を購入することで、自宅でもコンピュータを利用できる。ノート機をSFCのWS環境と近づけるためにUNIX互換コマンドなども用意したが、オペレーティングシステムの違いにより同一のものをノート機上で実現することができず、学生に混乱が生じた。また、各教室にはイーサネットが敷設されているが、今まで利用してきたノート機では直接接続することはできない。CNS

†Implementation and Evaluation of Mobile Computing Environment, by Kazunori Sugiura, Kimio Ishii(Faculty of Environmental Information, Keio University), Jun Murai(Keio University)

‡uhyo@wide.sfc.keio.ac.jp

§ishii@wide.sfc.keio.ac.jp

¶jun@wide.sfc.keio.ac.jp

にノート機を接続する時には、通信ターミナルソフトウェアによって、RS232Cターミナルサーバ経由でアクセスする必要があり、TCP/IPプロトコルを用いて直接CNSにアクセスすることは不可能である。さらに、MSDOSはテキストインターフェースを前提としたOSであるため、GUIとしてXウィンドウを利用しているSFCの環境とは整合性がとれなかった。

MSDOSを採用した場合の問題点の一つに、オペレーティングシステム自体のソースコードを始め、さまざまな仕様が公開されていないことがあげられる。SFCのWS環境は主にフリーソフトウェアから成り立っており、アプリケーションのソースコードを入手し、改良して公開することによって、よりよい教育や、環境を目指すことが可能である。しかしMSDOSでは、このようなことを実現するには困難が伴う。

またSFCでは語学教育にコンピュータを利用しているため、ノート機上でも韓国語、中国語、ロシア語、スペイン語、ドイツ語、フランス語等を扱えることが望ましいが、MSDOSの上では不可能であった。

## 2 目的と方針

本研究開発の目的は、キャンパスコンピュータ環境と完全に連続する可搬のコンピュータ環境を確立することにある。具体的には、キャンパスコンピュータ環境とは、X WindowをインターフェースとするBSD UNIXとインターネットプロトコル体系をベースにしたローカルエリアネットワーク上の分散システムであり、本研究で対象としている可搬コンピュータは、インテル386アーキテクチャによるノートブック型コンピュータである。

したがって、この目的を成就するためには、次の各課題を解決する必要がある。

- i) ノートブック型コンピュータ上の制限された資源でのBSD環境の稼働。
- ii) X Window および 各主要応用技術の移植。
- iii) システム起動と終了のオーバーヘッドの短縮。
- iv) キャンパスシステムとの連続性を実現するためのネットワーク技術。
- v) キャンパスシステムと整合したシステム管理機能。

ノート機は固定機と比べるとメモリ容量、HDDの容量、ディスプレイの大きさといった仕様面で制

限があり、従来のソフトウェアをそのまま移植しても、必ずしも満足に動作させることはできない。そこで、上記の問題に次のような方針で対応を行なった。

i) に関しては、University of California, Berkeleyで開発されているBSDオペレーティングシステムNet-2(Berkeley Network Release 2)のインテル386用のコードとそれを基盤にしたBSDI(Berkeley Software Distribution Inc.)社のBSD/386を基盤に、メモリやHDDなどの周辺機器に対応する拡張を行なった。

ii) では、グラフィックボードへの対応を行ない、低い解像度に対応した応用の移植と開発を行なった。携帯性が重視されるラップトップ、ノート機コンピュータは入力デバイスが犠牲になりやすい。ここでは、Xウィンドウを利用する場合のポインティングデバイスにはSFCのWS環境で通常利用している入力デバイス(マウス等)を採用している。

また、iii) は、レジューム機能の開発を中心に対応し、iv) は、作業空間の連続性と移動透過性の機能の確立を目指し、v) のために、セキュリティに関する検討に焦点をあて、実験を開始している。

## 3 ワークステーションとノートブック

ノート機はSFCが開校された時点で、各メーカーがその第1世代マシンの開発を終了しており、SFCでも推奨機種として利用してきた[4]。UNIXを搭載した携帯型コンピュータは、当時SPARC-LT[3]やNEWSなどがあったが、いずれも学生が購入するには値段が非常に高く、携帯性も低いものであった。しかし、ノート機は年々進化し、価格も安くなり、日常利用しているWSとの性能差も小さくなった。CPUの処理速度が上がり、小型で大容量のHDD、大容量の主記憶を持ち、なおかつ軽量化されたノート機は、現在SFCで利用している固定機と性能上ではほとんど変わりが無い。表1にSFCで学生が主に利用しているWSとラップトップ型WS、そして、インテル386アーキテクチャのノート機の仕様を示す。

また、表2にSFCで学生が日常利用しているPOP-NEWSとの性能比を示す。ノート機については、J3100SL 001VWを対象とした。ベンチマークにはBYTE UNIX Benchmarks (Version 3.11)を使用した。POP-NEWSは学生が通常利用している環境でベンチマークを起動させ、ノート機はローカルディスク上で行なった。処理速度的な問題よりも使い勝手の良さを考えている。

表 1: SFC の代表的 WS とノート機の仕様

項目	WS	ラップトップ	ノート機
モデル	PWS-1560	NNWS-1230	J3100SL
CPU(MHz)	48633-25	48633-25	801V W
FPU	48681	48681	386SL-25
主記憶(MB)	8	8	31781
副記憶	1024x768	1024x768	640x480
HDD(MB)	100	240	(120)
文字数	355x100x341	348x98x418	297x210x46
重量(Kg)	8.10	9.8	W2.2+HDD

表 2: SFC の代表的固定 WS とノート機の性能比

テスト項目	固定機	ノート	性能比
Dhrystone2(lps)	5755.7	5890.7	1.02
Dhrystone2+reg(lps)	6192.5	5881.2	0.95
int(lps)	1289.6	2092.7	1.62
long(lps)	1289.6	1994.8	1.55
float(lps)	318.6	890.4	2.79
double(lps)	297.4	852.6	2.87
Systemcall(lps)	2397.8	2504.6	1.04
Pipe Throughput(lps)	720.2	953.3	1.32
Context Switching(lps)	664.5	505.6	0.76
Process Creation(lps)	159.5	32.7	0.21
Exec Throughput(lps)	93.2	27.7	0.30
File Read(30Sec)(KBps)	1335.0	5339.0	4.00
File Write(30Sec)(KBps)	200.0	266.0	1.33
File Copy(30Sec)(KBps)	139.0	285.0	2.05
Shell scripts(8)(lpm)	1.0	1.0	1.0

この表が示す通り、性能面ではノート機は SFC の教室で用いられている WS とほとんど変わりがない。

### 3.1 ノート機と BSD/386

これまで SFC で推奨されたノート機は表 3 に示すように全て MSDOS 用であり、キャンパスの WS 環境とはかなり異なっていた。そこで WS 環境に合わせた MSDOS 用のアプリケーションを作成し、学生に配布していた。

表 3: 年度別推奨ノート機及びその性能

年度	機種名	CPU	HDD	OS
1990	J3100SS	8086	—	MSDOS3.3
1991	J3100SX	386SX	—	MSDOS3.3
1992	J3100V	386SX-20	40M	MSDOS5.0/V
	PS55 note	386SX-16	40M	IBMDOS5.0/V

SFC では、BSD/386 を “One Year University Licence” で購入し、学生・教職員は、キャンパス内及び自宅での利用が可能となった。また全てのソースコードが利用でき、研究・教育のための OS としても適切である。

BSD/386 が利用に必要な環境を表 4 に示す。

表 4: BSD/386 の動作条件

	サポート体制	備考
CPU	386SX,386,486	
FPU	80387	
BUS	ISA 及び EISA	MCA も対応予定
主記憶	4MB 以上	X 利用時 8MB 以上
HDD	IDE,ESDI,SCSI	80MB 以上
Display	各種対応	高解像度も対応
Ethernet	各種対応	

上記の動作条件から、本研究開発の対象としている仕様は概ね表 5 に示されるような性能である。

表 5: 対象とするノート機の性能

CPU	メモリ	HDD
386SL-25	8MB	120MB
486SX-25	8MB	120MB

## 4 キャンパス環境とノート機との整合性

ノート機に BSD 系の UNIX を採用することで、SFC の WS 環境との整合性は従来に比べると飛躍的に向上し、利用者の混乱も減ることになる。しかし一方では UNIX 特有のシステム管理が必要となる。

ノート機の GUI には、X ウィンドウを採用し、SFC の WS 環境と同一にする。仮想漢字コンソールを追加することで、X ウィンドウを利用しなくても日本語環境が利用できる(表 6)。図 2 に本実験のシステムを示す。

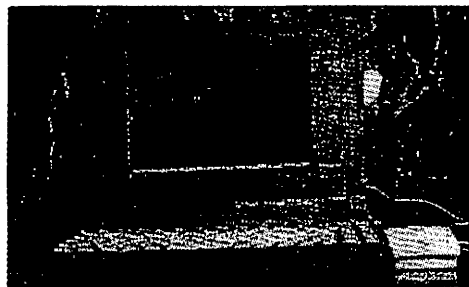


図 2: 本実験の対象ノート機

ノート機のネットワーク接続は、プリンタポートに接続する汎用のイーサネットアダプタや、専用の ISA 拡張ボックスにイーサネットカードを接続する

表 6: ノート機のソフトウェアアーキテクチャ

X アプリケーション 多言語環境 X ウィンドウ
仮想漢字コンソール BSD/386
ノートブック

ことで可能となる。移動ホストへの対応は第 5 章で述べる。

また、利用者一人一人が、各々のマシンを管理することになる。しかし特権利用者の権限をすべての利用者に与えることは、ネットワーク環境のなかでのセキュリティに問題がある。これらについては第 7 章で述べる。

#### 4.1 アプリケーション

SFC の WS 環境は、基本的に 4.3BSD 以降の BSD UNIX で統一されている (4.4BSD アーキテクチャの導入も試験的に行なわれている)。学生は情報処理言語 I を履修することによって、SFC の WS 環境を利用できるようになる。表 7 に学生が利用できるソフトウェアを示す。日常的に使用している環境は、表 7 のフリーソフト、PDS 等を基にして構成されているため、ノート機でも、最低限これらのアプリケーションは動作する環境が望ましい [8]。しかし、ノート機においては、限られたハードディスクの容量を有効に利用するために、学生の利用するアプリケーションの選択は慎重に行なう必要がある。また、アプリケーションによっては、ノート機の内蔵 HDD にインストールせずに、ネットワークに接続した時に利用できるように CNS のサーバに入れておくことも可能である。

表 7: ノート機的环境

項目	ソフトウェア
ウィンドウ	X Window System (X11R5)
シェル	tcsh, bash, zsh など
エディタ	NEmacs, mule, vi など
仮名漢字変換	Egg + Jserver, canna, kinput
電子メール	mh / NEmacs など
電子ニュース	gnus / NEmacs など
文書処理	TpX / TpX fig tgif
言語処理	C, Lisp, Scheme, Perl, Prolog

ノート機はその大きさの制約上、画面の表示範囲が狭い (640×400 ~ 640×480)。そのため、X ウィンドウ上で動作するアプリケーション、特に文書処理や作画処理に利用されるアプリケーションの多く

は、この表示範囲内には入り切らない。よって、アプリケーションの移植には工夫を要する。

#### ● X ウィンドウ

X ウィンドウには、白黒サーバ・カラーサーバの 2 種類を用意した。白黒サーバは、800×600 の表示範囲 (実表示画面は 640×480) を持つ。カラーサーバは、640×400 に表示範囲が限定されるが、256 色の同時発色が可能であり、グラフィックアプリケーション等の実行には適している。

X ウィンドウの描画速度は、コプロセッサをノート機に搭載することで、SFC で学生が利用している LUNA DT-20 相当の速度まで上げることができる。IBM-PC 互換機に対応させるという性質上、X サーバはある程度汎用性が必要になる。

tvtwm による仮想ウィンドウマネージャをノート機上で利用することによって、仮想的に大きな画面を利用することが可能である。仮想画面内は、Virtual-Desktop 上をマウス、キーボードで操作し、利用することができる。キャンパス内で利用されているウィンドウマネージャである twm がベースになっている。

T<sub>E</sub>X のプレビューである、xdvi は、表示フォントを可能な限り小さくし、部分拡大機能によって細部の表示に対応させた。グレイスケールによるスムージングによって、ドットの荒れを滑らかにした。図 3 にそのスクリーンダンプを示す。グレイスケールによるスムージング効果のため、この例では xdvi が見にくくなっている。

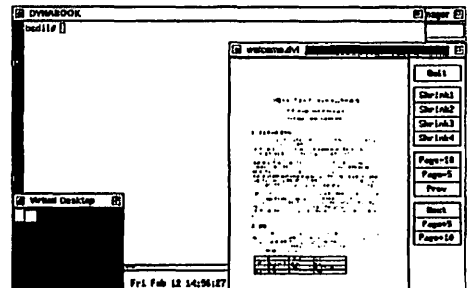


図 3: ノート機上の X ウィンドウとプレビュー

#### ● 仮想漢字コンソール

X ウィンドウを利用しない場合でも、仮想漢字コンソールによって複数のコンソールを持つこ

とが可能である。漢字ドライバは、386BSDの KanjiHand Ver 1.0 を基にした。仮想コンソールドライバはカーネルに含まれ、キー操作で切替を行なう。コンソールの漢字フォントは、ノート機に搭載されている漢字 ROM のデータ、X ウィンドウ用の bdf フォント、MINIX-V の public font のいずれかを利用できる。

通常のコンソールは CGA 画面が基となっているため表示文字数が 80×25 の 2000 文字が限界となるが、仮想漢字コンソールは VGA のグラフィック画面を利用することで、640×480 上に任意の大きさのフォントを選択することが可能となり、利用するフォントの大きさによって最大表示文字数も増やすことができる。通常の英語コンソールと比較しても、表示速度の差はあまり感じられない。

漢字コードは、SFC の WS 環境と同一の JIS コードに対応しているため、従来の MSDOS 環境のノート機を使用していた時は漢字コードの変換が必要であったが、仮想漢字コンソールの採用によってこのような煩わしさは解消される。

仮想漢字コンソールを利用することで、X ウィンドウのアプリケーションをディスクにインストールしない場合は、ディスク容量の節約にもつながる。

多言語表示対応コンソールは、現在開発中である。基本的に mule を起動した後、自動的に多言語環境に移行するよう配慮する。将来的には現在様々な機関で検討されている多言語コード体系にも対応する予定である。

#### ● レジューム機能など

現時点では、レジューム機能を有効にするにはログアウトする必要があり、アプリケーションを動作させたままの状態でのレジューム機能は、今後の課題となっている。またシャットダウン時には、自動的に電源まで切られるようになっている。バッテリーのみでノート機を利用する場合を考えて、スリープ機能も設けた。これはアイドルを自動的に検知し、バックライトを消す機構である。

#### ● MSDOS との共存

ノート機の HDD には MSDOS の共存が可能である。物理パーティションを切り分けて 1 台のハードディスクで利用できる。事前に MSDOS 用の起動パーティションを用意するかどうかについては、検討中であるが、120MB 以上の

HDD を利用する上では可能となる。UNIX では利用できないアプリケーション、ゲーム等は、起動パーティションを MSDOS に切替えて使用する。また BSD/386 内からも MSDOS エミュレータによる、エミュレーションが可能であり、実用的な処理速度は望めないながらも利用可能である。

#### ● その他

ノート機を学生がある程度保守、管理できる簡単なコマンドも複数用意した。これは、あまり UNIX に詳しくない学生でも利用できるように、GUI を利用したい。

## 5 移動ホストへの対応

ノート機の可搬性を活用する為には、少なくともキャンパス内ではどこからでもネットワークに接続できることが必要である。しかし現状の TCP/IP のネットワークでは、ノート機のような移動ホストには対応できない。インターネット上では移動ホストをサポートする方法は様々なものが検討されているが、その中の VIP (Virtual Internet Protocol) [10] を実装し、現在実験を行なっている。VIP は ID としての仮想的な IP アドレスと、実際に接続している場所の IP アドレスを組合せて、移動ホストに対する通信を実現している。さらに、DHCP (Dynamic Host Configuration Protocol) [6] を用いた IP アドレスやサブネットマスクなどの情報を動的に設定する機構を作成中である。DHCP サーバが完成すれば、ノート機はイーサネットケーブルを接続するだけでキャンパスネットワークにアクセス可能となる。この二つを組み合わせることによって、キャンパス内はノート機を自由に接続できる。

VIP を用いて移動ホストが通信する具体的な仕組みは以下の通りである。

1. 移動ホストはネットワークに接続すると、DHCP サーバにアドレスの割り当てを要求し、一時的なアドレスを獲得する。
2. モービルマネージャと呼ばれる移動ホストの親ホストに、接続を知らせる。この時、途中のゲートウェイが VIP に対応していれば、キャッシュに登録される。
3. 通常のホストから移動ホストへの通信は、モービルマネージャに向けてパケットが送られるが、経路上にキャッシュがあれば、その登録されている一時的な IP アドレスに送られる。

4. VIP に対応するゲートウェイが一つもない場合にも、モバイルマネージャが移動ホストの一時的な IP アドレスにパケットを送る。

現在、経路の冗長性に起因するオーバーヘッドを含めた定量的な評価を行なっている。CNS におけるホームディレクトリとノート機上のホームディレクトリを融合させるような仕組みが必要であると考え、現在 coda ファイルシステム [5] のような分散型ファイルシステムを調査検討中である。

## 5.1 イーサネットの接続

イーサネットで、ノート機を接続した場合、固定機に比べて転送速度にどれぐらいの差が生じるかを表 8 に示す。比較機種は固定機 A が LUNA DT-20、固定機 B が NEWS-1560 の 2 機種で、ノート機は、J3100SL 001VW に、プリンタポートに接続する小型汎用イーサネットアダプタ (Xircom) と、ISA バスに接続する拡張イーサネットボード (we0) を試した。

表 8: 固定 WS とノート機のイーサネット転送速度の違い

項目	固定機 A LUNA	固定機 B NEWS	ノート we0	ノート Xircom
速度 (KB/s)	89	160	89	19

小型汎用イーサネットアダプタは、携帯性が高いが転送速度は低い。拡張イーサネットボードは、十分な転送速度を得られている。このことを考えると、小型汎用イーサネットアダプタを利用する場合は、必要最小限のトラフィックにする必要がある。SFC の環境を考えると、小型汎用イーサネットアダプタを利用した方が、キャンパス内では接続場所が限定されないため有利であると考えられる。

## 6 ノート機の効果的な資源利用

高性能なノートブックを学生が手にするのは、現時点では価格の点で難しい。特にメモリ・HDD の容量が限定されるという問題があり、またそれらは容量に比例して価格が上昇する。そのために限られたメモリ・HDD 容量で実用的な環境を構築する必要がある。

主記憶容量が少ないと、ページングによるスラッシングが発生する。特に大きなメモリを必要とする jserver や X ウィンドウを利用する環境を考えた場合、どれぐらいのメモリ容量が必要となるか、ノ

ート機を起動したときの時間 (時間 1)、ログイン後の X ウィンドウの起動時間 (時間 2)、ならびに vmstat -s による swap-in と swap-out の量を測定した (それぞれ、swap1、swap2)。

表 9: 実用的な主記憶の容量

主記憶	時間 1 秒	時間 2 秒	swap1 (in,out)	swap2 (in,out)
2MB	136	—	1477,1432	—, —
4MB	74	93	538,586	2218,1989
6MB	57	58	443,0	1267,1025
8MB	58	55	434,0	1160,948
10MB	56	46	433,0	1030,850

まず主記憶 2MB ではスラッシングが発生し、X ウィンドウは 10 分を経過しても起動されなかった。また表 9 からわかるように、主記憶 6MB 以上では、起動時に swap-out は起こらなくなった。よって実用的な環境を構築する場合、主記憶は 6MB が最低限必要であると考えられる。X ウィンドウのユーティリティを活用する場合は、主記憶を 8MB は確保したい。

HDD のパーティションについては、搭載するメモリ容量と学生が通常利用するソフトウェアを考慮して、/ ファイルシステム、スワップエリア、/usr ファイルシステムの 3 つのパーティションに分割した場合の最適値を算出する。この最適値を表 11 に示す。メーカーからノート機用に販売されている HDD の容量には、40, 60, 80, 120, 200MB があるが、実用性を考えると 80MB 以上が必要である。必要なソフトウェアの容量を表 10 に示す。なお、ノート機上で学生が利用するホームディレクトリは /usr/home のしたに設定される。単純に /usr パーティションから、アプリケーションで利用している容量を引いたものをホームディレクトリの最大容量に設定することは、管理上好ましくないため、soft quota、hard quota を設定して対応する。CNS では学生のディスク使用量がログイン時にアプリケーションによって表示される。これと同様の仕組みを用意することで、利用者の混乱を避けることができる。

120MB 以上の HDD を搭載させる場合、前述したように起動パーティションを区別することによって BSD/386 と MSDOS の両方を一つのディスクで共存させて使用することが可能になる。

### 6.1 圧縮型ファイルシステム

現在、ノート機用の圧縮型ファイルシステムを作成中である。圧縮型ファイルシステムは、基本的に UNIX のファイルシステムのデータブロックに格納される部分を圧縮し、HDD の容量を節約する。圧

表 10: 必要なソフトウェアの容量 (MB)

パッケージ	容量
/ ファイルシステム (基本)	3.7
/usr ファイルシステム (基本)	14.0
C コンパイラ環境	6.8
ネットワーク環境	3.1
マニュアル	3.0
X ウィンドウ	19.0
SFC 環境ユーティリティ	23.0
合計	72.6

表 11: ノート機の HDD パーティション構成

	80M	120M	200M
/	6.6M	8.5M	8.5M
swap	10M	17M	24M
/usr	63.4M	(94.5M)	(167.5M)

縮・展開は、自動的に行われるが、ユーザの意志で圧縮を行わないことも可能である。なお、マニュアル、ライブラリ、ヘッダファイルなどは、あらかじめ圧縮された状態でインストールされていて、必要に応じてソフトウェアが展開する。圧縮比率は、時間と容量のどちらを優先させるかによって選択が可能になる。

## 6.2 共有ライブラリ

主にノート機など、容量が限られた HDD を使用せざるを得ない場合、共有ライブラリは、有効なディスク資源の確保へとつながる。現在のところ BSD/386 は、通常の静的にリンクされるライブラリを使用しているが、将来的には共有ライブラリへと移行することは間違いない。

共有ライブラリがどの程度ディスク容量を節約するかは、表 12 で示されている Linux 上で実現されている共有ライブラリによって作成された X ウィンドウユーティリティのディスク使用量から推測できる。共有ライブラリを使用する場合、ライブラリのバージョン管理が問題となるが、ノート機を利用する学生に意識させることなく、バージョン管理を可能とする仕組みを用意する必要がある (後述)。

## 7 管理運用技術

学生が一人一台利用する場合、管理やセキュリティ上の問題が生じる。UNIX は、常に管理者と利用者が区別されている OS であり、キャンパス管理システムとの協調が必要となる。

表 12: 共有ライブラリ利用時におけるディスクの使用量

	通常のライブラリ BSD/386	共有ライブラリ Linux	比率
X	1,155,072	947,818	0.82
bitmap	606,208	70,860	0.12
twm	388,174	111,620	0.29

### 7.1 newuser 登録プログラム

ノート機には、newuser というユーザ登録プログラムがインストールされ、学生が初めて電源を入れた時に自動起動し、所有者のアカウントを作成する。本人であるかどうかの確認は、誕生日、学籍番号、あらかじめ SFC で与えられたログイン名などを参照して行なう。同プログラムは、アカウント作成終了後消去され、以後ノート機は所有者以外がログインすることはできない。管理者用の root のパスワードもあらかじめセットされ、基本的に学生には教えない。シングルユーザモードでの起動時も root のパスワードが必要になるよう設定を変更する。

キャンパスのネットワークを破壊しようという悪意を持つ学生はいないであろうが、root になれるマシンが数多くあるというのは、ネットワークにとって非常に危険なことである [7]。熱意と技術のある学生にとっては自由度が小さくなるが、セキュリティの方がより重要であると判断した結果、上記のような仕組みを取ることにした。

### 7.2 ソフトウェアの保守管理

ノート機の保守には、ソフトウェアのバージョン管理も行わなければならない。SUP[9] や, rdistなどを基にしたバージョン管理プログラム及びサーバ (Notebook Server (NS)) を用意し、自動的に行なうようにする。NS デーモンはノート機が CNS に接続された時にソフトウェアのバージョンを調べ、NS に直接アクセスすることによって、場合によっては古いものを新しいものと取り替える。具体的にはノート機はあらかじめインストールされているソフトウェアのリストを、決められた管理用ディレクトリに持っていて、このリストと照合することによってバージョン管理を行なう。

## 8 おわりに

可搬のノート機と一般の分散型キャンパスコンピュータ環境との完全な整合を目的とした実験運用を慶應義塾大学湘南藤澤キャンパスの環境で開始し

た。オペレーティングシステムを含めた主な機能の移植、限定された資源への対応、移動ノードへ対応する通信技術、そして、管理運用システムとの整合をほぼ完了し、実用に十分な評価を得ることができた。

ノート機とキャンパス用分散環境の融合には、まだ多くの課題がある。特に、移動ノードとして動的に接続性が変化するなかでの、分散システム上の役割に関しては極めて興味深い議論と検討の課題がある。また、個人所有の権限の中でのセキュリティを中心とした管理運用問題、障害とその影響に関する問題、接続設備、充電設備、配電設備といったキャンパスの設備上の問題は実用に向けて重要である。

本実験は現在は限定された実験運用を行っており、1993年度から拡大実験運用を開始する。

## 9 謝辞

常に相談に応じてくれた共同研究者の方たち、特に慶應義塾大学・環境情報学部村井研究室の登坂 章弘氏、林 周志氏、徳田研究室の冨永 明宏氏、小山高明氏に感謝します。論文作成及びデモ作成に関する相談にのっていただいた、加藤 朗氏、楠本 博之氏、徳田 英幸氏に、感謝します。多くの共有可能ソフトウェア開発に努力をされている方、特に、カリフォルニア大学バークレイの CSRG とその周辺のグループ、ここで用いた数多くの日本語環境/多国語環境の確立に具体的貢献をされている方々に感謝します。BSD/386 を快くリリースしてくれ、多くの要求を聞き入れて開発の協力をしてくださった Rob Kolstad 氏、Rick Adams 氏、ならびに BSDI 社の皆様に感謝します。

## 参考文献

- [1] 安村通晃、有澤誠、斎藤信男. Keio sfc におけるコンピュータリテラシー教育. 情報処理学会 第 32 回プログラミングシンポジウム, January 1991.
- [2] 加藤朗、楠本博之. 慶應義塾大学湘南藤沢キャンパスのネットワーク構築. 日本 UNIX ユーザー会第 17 回 UNIX シンポジウム, July 1991.
- [3] 村井純、松山直道. ポータブル UNIX ワークステーション SPARC LT. ASCII, 1991.
- [4] 中森智博 高野雅晴 多田和市, 横田英史. 携帯型パソコンの将来像を探る. 日経エレクト

トロニクス, No. 496(4-2):pp.95--120, 1990.

- [5] Satyanarayanan M., Kistler J.J., Kumar P., Okasaki M.E., Siegel E.H., Steere D.C. Coda: a highly available file system for a distributed workstation environment. *IEEE Transactions on Computers*, 39(4), April 1990.
- [6] R. Droms. *Dynamic Host Configuration Protocol*. Bucknell University, December 1992. INTERNET-DRAFT.
- [7] Simson Garfinkel and Gene Spafford. *Practical UNIX Security*. O'Reilly & Associates Inc., 1991.
- [8] SFC-CNS ローカルガイド編集委員会. *SFC-CNS ローカルガイド*. 慶應義塾大学湘南藤沢メディアセンタ, 1992.
- [9] S. A. Shafer. *The SUP Software Upgrade Protocol*. CMU Computer Science Department, 1985. man page of the sup command.
- [10] Fumio Teraoka, Kim Claffy, and Mario Tokoro. Design, Implementation, and Evaluation of Virtual Internet Protocol. In *Proceedings of the 12th International Conference on Distributed Computing Systems*, June 1992.