



電氣通信大學MMA

## CONTENTS

1	部長の言葉	3 C	村上敏之
2	UNIXとMMAの関係と歴史	M 1 J	橋岡孝道
4	TRONプロジェクトについて	3 C	村上敏之
6	赤外線による伝送実験	3 J	今井潔
8	サーバって何？	4 C	u n c o v e r
10	Internetには猿が棲む	2 C	r i k a c h a n
16	Xウインドウプログラミング p u s h m e の製作	1 E	ういろう
17	Lecture Schedule Version 1.2.1 for UNIX	1 J	S o - M i y a
22	スーパーパーユーザの憂鬱	1 J	佃良生

表紙 1 J 鈴木数代

# 部長の言葉

～Micro-computer Making Association (MMA) のコーナーへようこそ～

電子情報学科3年 村上敏之  
(E-mail: e11163@ced.cas.uec.ac.jp)  
(MMA local:p2)

平成6年11月8日

本日はMMAのコーナーへ起こしいただきいただきまして、まことにありがとうございます。

当サークルは、もともとコンピュータを自作するサークルだったのですが、CPUをはじめとする半導体の発達により、アマチュアが最先端のマシンを作る事はほぼ不可能になってしましました。そこで現在は、市販ハードを用いた、ソフトウェアに重点をおいた活動を行っています。もっとも、市販ハードでもMMAに来たら最後、ジャンクと化すのが落ちなのですが。

そのためか、メインマシンで使っていたSCSIのハードディスクが、「こんなに人（？）使いの荒いサークルはいやだ」と思ったのか、家出しまったのです。部員が作ったデータやプログラムを持ったまま。飼い主達（部員）がよっぽど気に入らなかつたようです。反省しなければ。

さて、今年の展示ですが、この原稿を書いている時点では、1年生諸氏が作成したプログラムを主に展示する予定です。それらのプログラムの幾つかについては、当誌に記事が載っているはずです。それもあわせてご覧ください。

# UNIX と MMA の関係の歴史

楯岡孝道 (tree)

tate@cs.uec.ac.jp

平成6年 11月 7日

## 1 はじめに

今や MMA は UNIX サークルかのように見える。事実、現在の活動のほとんどは UNIX を用いたプログラム開発になっており、UNIX と MMA は切っても切り離せない関係となっている。

しかし、Micro-computer Making Association という名前に “UNIX” の文字が無いことからもわかるように、本来 MMA と UNIX は強く結びついたものではない。

既に UNIX の無い MMA を知らない部員が殆どとなり、この単純な事実を伝える者が減りつつある今、部室で UNIX が起動した時から MMA と UNIX との関係を見てきた者として、歴史を振り返りこの記録を残す。

なお、筆者自身が UNIX と共に歩んだ世代であるため、UNIX 導入以前の MMA に関して誤解がある可能性が否定できないことをお断りしておく。

## 2 石器時代

初期の MMA では、利用できる計算機に不満が多く、この不満を解消するためにハードウェアの自作が盛んであった。

当時の計算機では CP/M などが動作し、8 インチのフロッピーディスクなどが稼働していた。また、SC/MP などの新しいプロセッサの導入

にも積極的であったようだ。

伝説として伝わるエピソードがいくつかあるが、既に意味のわかる者は少ない。本題から外れるのでキーワードを 2 つ程あげておくに留める。

- トレーナーで bootstrap
- 季節を感じる floppy

## 3 Before UNIX

MMA で製作されるマシンのプロセッサとして MC68000 が多用されるようになると、マルチタスク/マルチユーザ環境をもった OS9/68000 が移植され動作を開始した。

OS9 は UNIX に近い機能を持ち、デバイスドライバが階層構造を持つなど UNIX より優れた一面もある OS で、部員の開発意欲をかきたてた。OS9 上では UNIX 上のツールなどの移植などが盛んに行われ、疑似 UNIX 環境とも呼ぶべき環境が整っていった。

疑似 UNIX 環境はシェルやユーザ認証機構にもおよび、1992 年まで動作していた “ARES” は 5 台以上の端末を接続し、コマンドレベルでは殆ど UNIX と区別がつかなかった。

計算機ネットワークの構築も計画され、無線モデムを用いた接続実験なども行われていたが、部室を襲った雨漏りにより部内の殆どのシステムが崩壊した。

また、"ARES" の後継メインシステムとして設計された "MILK" の為の OS9/68020 のライセンス料が非常に高価で、結局購入できなかつた事などにより、OS9 を使ったシステムは新規に開発されなくなつた。

## 4 Mach386 の誕生

1991 年に部室に IBM-PC/AT 互換機が持ち込まれた。16MHz で動作する i386SX プロセッサに 8MB のメモリという貧弱な構成であったが、当時部内で唯一動作していた "ARES" よりも数倍高速な処理速度は部員を魅了した。これがその後数度のアップグレードを経て部内のサーバマシンとなる "beat" である。

この AT の為にこの年、米 mt Xinu 社より Mach386 オペレーティングシステムを輸入し、本格的に部内で UNIX 環境が動作をはじめた。Mach386 は Mach 2.5 をベースにしており、システムコールレベルで UNIX との互換性がある上 X11R4 も付属していたため、UNIX 用フリーソフトウェアの移植や X window のプログラム作成などが盛んに行われた。

当然ネットワークに関しては Ethernet と SLIP がサポートされていたため、10base2 ケーブルを用いた他の PC との Ethernet 接続や、MS-DOS 上で動作する KA9Q との SLIP 接続実験なども行われ、部員の間にネットワークに関する知識と経験が蓄積された。

Mach386 は比較的初期にリリースされた i386 アーキテクチャ用 UNIX という事もあり、ライブラリやコマンドなどに多くのバグが存在した。MMA の部員はこれをかえって楽しみ、それらの問題を解決していった。

## 5 NET/2 の晚餐

Mach386 のバグはカーネルにまで及んだ為、バイナリライセンスでは対処できずに放置する事が増えてきた一方で、NET/2 を起源とする 386bsd がリリースされた。

初期の 386bsd はその不安定さと導入の難解さで敬遠されたが、パッチキット等がリリースされ当時の MMA の技術力でも導入が可能になつた頃、計算機システム内に手の出せない所が存在する不快さを解消する物として、新たに導入された。

その後、386bsd の後継である NetBSD や、他の NET/2 の系列である BSD/386、独自に書き下ろされた Linux など PC-UNIX と呼ばれる OS がリリースされ、利用可能になるたびに OS を入れ替え、速度や使用感などを評価する日々が続いた。

この間 "beat" はマルチ I/O カードとイーサネットカード以外は全て交換される程のアップグレードを繰り返し、部内のサーバマシンの地位を固めていった。

## 6 天変地異を経て

MMA の UNIX マシンはディスククラッシュを何度か経験している。また、それ以外の原因によってもディスクの内容が全て失われた事もある。

それでも MMA 部員たちは各種環境構築を止めようとはしない。

そして恐らくこれからも止めないであろう。しかし、その環境として UNIX が選ばれ続けるとは限らない。部員にとって UNIX は「快適な計算機環境を構築する」為の手段に過ぎないからである。

# TRONプロジェクトについて

電子情報学科3年 村上敏之  
(E-mail:e11163@ced.cas.uec.ac.jp)  
(MMA local:p2)

平成6年11月8日

あなたはTRON(トロン)プロジェクトを知っていますか?  
知らない?では、そんな人のためにちょっと……  
ん?

「知ってる。何年か前にだいぶ流行ったけど結局普及しなかった」と  
思った人もいるようですが、それは事実誤認ですので以下の記事をお読み下さい。

TRONは10年前(1984)、東大の坂村健助教授(当時助手)が始めたもので、現在日本で行われている最大のコンピュータプロジェクトです。

このプロジェクトは、近い将来、身の回りのあらゆる物にコンピュータが内蔵される時代がくる。そのときに操作方法やデータ形式がばらばらでは困る、という事で始まったそうです。また、そんな時代に基礎的な技術情報を独占すると、社会の発展に差し障りがあるので、TRONの仕様は誰でも仕様使用料を払わずに使えます(例えばUNIXだと、AT&Tにライセンス使用料を払わなければなりません)。

TRONは幾つかのサブプロジェクトに分かれています。何故かといふと、例えば車だと、F1で使用される車とトラックとでは、車は車でもでも異なる物になります。コンピュータでも同じ事がいえて、スペースシャトルの制御に使われるコンピュータのOSとパーソナルコンピュータ用のOSとでは、要求されるのはまったく違うからです。

では、サブプロジェクトの中から幾つかを紹介していきましょう。

○ T R O N c h i p

日本独自の C P U。東芝、日立、富士通、三菱、沖、松下の6社が開発している。特に、日立が開発した Gmicro/500は、同一クロックの場合インテル社のペンティアムよりも速い。

○ I T R O N

Industrial TRONの略。

主に機器に組み込んで、それを制御するために使う。

小型軽量だが、完全なマルチタスクなりアルタイム O S。

○ C T R O N

Central TRONの略。

主に電話交換機など、非常にたくさんのタスクの管理を要求され、なおかつリアルタイム処理でなければならないものに使われる。

○ B T R O N

Business TRONの略。

パーソナルコンピュータ／ワークステーション用の TRON。

体が不自由な人でも扱えるような工夫が、O S レベルでなされている。GUI を使用しているが非常に軽い、完全なマルチタスク O S である。

このように T R O N はリアルタイム・完全なマルチタスクである事が特徴です。

ところで、T R O N は身の回りにかなり浸透してきています。

例えば、電話をかける時は、何をどう頑張っても電話交換機の世話をになります。（系電話なら話しは別ですが……）

じつは、その電話交換機（電子交換機）用の O S として、N T T が C T R O N を標準としているのです。それどころか、A T & T やノーランテレコムなども採用を決定しているので、電子交換機用の O S として世界標準となりつつあるのです。

また、自動改札機やテレビ、コピー機などを代表とした機器制御／機械制御の分野では、I T R O N が事実上の業界標準 O S となっています。

このように、T R O N が普及しなかったというのは、嘘ですし、プロジェクトは終了してもいません。これからが正念場なのです。

# 赤外線による伝送実験

今井 潔

平成6年 11月 8日

## 1 はじめに

MMAをIP接続するために、これまでさまざまな提案をしてきました。今回の実験もその一環で、赤外線を使っての信号の空間伝送について考えてみました。

## 2 赤外線になった経緯

ご存じの通り、これまでMMAはいろんな手段をこうじて他の建物に手足を延ばそうとしてきました。その理由は学内ネットワークにIP接続するためでした。以下にこれまでなされた提案を示します。

### 2.1 10base2による部室-G棟直結作戦

ほしゃった理由:一番初めはこれでも案外実行可能な案にみえました。たしか、部室-G棟間が約66mで部室の中のネットワークの長さを入れても150m以内におさまる計算になりました。gatewayとしてはピザボックスに入れた386を使い、G棟に置くというものでした。しかし、この案は、もし部室-G棟を結ぶケーブルに雷が落ちた場合両端のコンピュータが黒こげに、最悪の場合そのネットワークセグメントに繋がっているコンピュータが全滅してしまうおそれがあり実行できませんでした。

### 2.2 無線によって接続する作戦

ほしゃった理由:まず、アマチュア無線による接続を考えました。巷ではよくパケット無線なるものをやっていますが、これと同じ方法です。しかし、公式にIPアドレスを取得する場合、アマチュア無線の規約に違反している送信内容のコード化(暗号化)がネックになるのではと危ぶまれました。そこで、ちょっとひよって微弱電波による通信を考えました。よく売られているFM送信機とFM受信機を使って電波法に触れない範囲の電波強度で通信をおこなうという

ものです。これは使う FM 送受信機をかなり精度のよいものにしなければならなかつたことと、送信電波と受信電波が混信しやすいという理由でだめになりました。

### 2.3 光ケーブルによって接続する作戦

ほしゃった理由: 上の経験を踏まえて考え出されたのがこの方法でした。有線による伝送経路の確実な確保と、確実な絶縁を考えるとこの方法がベストの方法に考えられました。幸い、部室に光モデムもあったのであとはケーブルを買うだけだったので当時の顧問の砂原さんにお伺いを立てたところ、どのような手段であっても建物間の結線は許さない、というお答えだったので即お蔵入りとなりました。

## 3 赤外線による部室 - G 棟間の接続

今回の案は、赤外線による建物間の接続です。空間を無線でつなぐもので、混信の可能性が少なく、手軽に扱え安全性の高いものという条件を満たしています。

### 3.1 赤外線による接続の実現性

赤外線というのは、リモコンなどによく利用されているあれです。リモコンでは、リモコン側についている赤外線 LED と家電側にある赤外線受光モジュールによってデータの伝送を行っています。原理的にはこの赤外線 LED と受光モジュールのセットを 2 組用意してそれぞれのコンピュータに付ければいいわけです。赤外線 LED も赤外線受光モジュールも秋葉原で手に入ります。また、赤外線 LED を 1-2A でドライブすると 32m 以上先でも信号を受光できるようです。

### 3.2 赤外線受光モジュールの信号フォーマット

それでは、単純に LED に 1200bps 程度の RS232C の信号を入力してやればいいかというとそうもないようです。受光モジュール側に問題があります。受光モジュールは、赤外線受光ダイオードとアンプ、フィルタなどを含む増幅素子からなり、PPM(Pulse Position Modulation- パルス位置変調) という変調をされた信号しか認識しません。具体的には、受光モジュールごとに決まっている変調周波数(33-40kHz) にあわせてパルス化された信号を 600us(=T) を受けとり、そのあとの無信号区間が T 程度なら 0、3T なら 1 となります。このことがネックとなり受光モジュールをもちいた通信では 300bps 程度の速度しか期待できません。

### 3.3 通信速度の改善

300bps 以上の通信速度にするのに、2つの方法を考えました。一つは変調周波数をあげる方法、もう一つは受光モジュール自分で設計してしまう方法です。今回の調布祭では、まず 300bps 程度の RS232C 用の回路を完成させ、その伝送実験とともに前者のほうの方法で通信速度の改善をはかりたいとおもいます。後者のほうも一応考えてはあります。赤外線受光ダイオードの出力を直接 OP アンプに入れて増幅します。信号の背景にはいるノイズは、ちかく置いた赤外線ダイオード(こちらには、信号がほいらないようにする)との差分をとることによって除去します。まあ、これがうまくいかず実験してみなければわかりませんが。

## サーバってなに?

uncover

平成6年 11月 8日

- rainbow が今日落ちた。分散処理はむずかしいなあ。
- snow-white のハードディスクがクラッシュした、モザイクのページも一緒に消えちゃった。

このように、現在の大半のコンピュータシステムはサーバと呼ばれる、\* ちょっとやー \* のマシンに非常に依存したものとなっています。このために、サーバマシンがダウンしてしまうと、それに仕事を頼んでいる他のマシンも一連托生、ダウンしてしまうのです。上の例ではファイルサーバと呼ばれる、大ディスク持つマシンがダウンしてしまったというわけですね。

このようなことがあってはいけない、そう思う人は多いはずです。そりやそうですね、一回上のようなことを経験してみれば、誰だってそう思うはずです。しかしながら、サーバマシンという考え方が非常に便利なことも確かです。もし、ログインするマシンが違うたびに、ホームディレクトリの場所も内容も違っていたら、これはとてもまだるっこしいことになるはずです。

このようなサーバマシンを持ったシステムは、ファイルの共有の他にもいろいろあります。そのシステムとうけられるサービスの具体例をあげてみましょう。

- NFS

他のホストにくつついてるディスクを自分のマシンにくつついてるように見せてくれるシステムです。本物のディスクを持っているのがNFSサーバとよばれているマシンであり、上の例ではrainbowやsnow-whiteがそうです。

- NIS

違うマシンでログインしても、ユーザ名、パスワードはいつでも同じですよね(たぶん)。これはNISサーバとよばれるマシンがそれらの情報を握っていて、問い合わせのたびに、教えてくれます。rainbowはNFSサーバの他に、このNISサーバもやってるらしいです。

- WWW

Mosaicという名前の方が通りがいいでしょうか。WEBサーバと呼ばれるマシンに接続して、ページのデータを転送してもらいます。なじみのあるマシンとしてはwww.uec.ac.jpですかね。

- anonymous ftp

だれでもftpができるように公開されているftpのホストですね。電通大ではftp.uec.ac.jpなんかがいろいろおいてあって楽しいんじゃないでしょうか。

- DNS

名前→IPアドレスの変換を行うサーバです。IPアドレスというのは、おおさっぱり言えば、コンピュータが他のコンピュータを識別するのに使うアドレスです。

ざっとみただけでも以上のサーバの考え方を用いたシステムがあるわけです。これらは非常に便利であることは間違いないのですが、やはりサーバがいる以上、上の例にあげられるような弱点があるわけです。これを克服するためにはサーバを多重化するといったアイディアがあります。これで大分、丈夫で元気なシステムにすることができるのですが、まだまだ丈夫さがたりないという人達がやっぱり世の中にはいて、彼らはサーバを無くそうと闘っている。この分野での研究は始まったばかりなので、まだまだ楽しむ余地がいっぱいあります。これを読んでいるあなたが、その新しいシステムを、MMAのコンピュータ上で実現してみてはどうでしょう？ MMAのコンピュータは学校の計算機室のそれとは違って、速度こそ遅いですが、やりたい放題の実験ができますよ。

## Internetには猿が棲む

rikachan@mma.club.uec.ac.jp  
hideo-m@tremaine.cc.uec.ac.jp  
hideo-m@atom.co.jp  
hideo-m@iij.ad.jp

Nov. 8 1994

### 1. はじめり

はじめりはxmosaicというあやしげなbinaryのprogramであった。ftp.iij.ad.jpの中を探検している中でふと見つけたこのプログラムは、それなりに面白かったが、個人的にはふーんと思っただけで終わってしまった。自分のhomeに置くにはやけにでかかったし、その当時はあまりその方面に興味はなかったのだ。第一そのソフトが何をするものなののかということさえ知らなかつた。

そのうちWorld-Wide Webなるもののブームの前兆が訪れはじめたころ、電通大でもserverをあげるからなんかやらないといわれて当時はmail addressが増えるのが嬉しくて安請け合いしたりした。

とりあえず何かやらにやあかんというのでWWWで使用する言語のHTMLの簡単な入門とかWWWとは何ぞや的な文章でごまかしていた。<sup>\*1</sup>

そんなある日に某tree先輩に連れて行かれたIIJで私のWWW環境は激変を遂げるのだった...

会った人物は30代前半ふうのあんちゃんが2人。これからWWWのserverを上げたいのでお前には技術的なことをやってほしいとのたまう。今ならこういう人は腐るほどいるけども、当時はこいつらまちかいなとしか思わなかつた。とりあえずその当時はア〇キーを逃げ出して<sup>\*2</sup>、金に困っていたのでとりあえず今の技術+先輩に聞きまくるでどうにかなるやねと思ってこの結構時給のいいバイトに飛びついたのだった。

てなわけで私もWWWのスペシャリスト(笑)として身を売ってはや半年ちょっと。ここでひとつ後ろでも振り返ってみましょうか。

### 2. server側から見るuser層

私は大抵の場合、一つktermを上げてtail -f /res/log/httpd-accessとかやっています。そうやってほけら~とlogを見て目立つuser層が分けてみましょう。

一見 ホームページだけちらっと見て帰る。

tail -fでログを見ているとこういう人が一番がっかりする。せめてどこがつまんなかったか言ってくれええ...

同業者 こいつらは以外にタチが悪い。 なにしろ内部データをごっそり見て行くのならともかく、定期的に  
— note —

\*1 一部ではこの当時の私の運営は非常に評判が悪かったらしい。

\*2 逃げ出したというより何も仕事をしていなかったので干されたというのが正しい。

\*3 Hyperlinkを終わりまで追い続けてひたすら内容をmirrorしようとする恐るべき回線キラー

robot<sup>\*3</sup>を動かしていたりするのだ。とくに富士通とかDECはひどい。毎日やってくるんだもんなあ。

エロ外人 「Sex」この3文字にしか興味のないバカ。

とにかくこいつらのバイタリティーと根気には脱帽。エッチはここまで人を変えるのね。トホホ。

ちょっと回線が重かったり、メンテで止めているとたんにメールが飛んでくる。そういう奴の決めゼリフはなぜかいつも「こういうImageをftpで公開しないの?」なんだよなあ。死んでもやらないから安心してくれ(笑)。

### 3. httpdを比べてみれば...

さてちょっとだけ真面目な話をしてみましょう。

現在UNIX上で動くhttpdの最大手はNCSA httpdとCERN httpdでしょう。<sup>\*4</sup>この2つを私の経験とカンで比べると...

・NCSA httpd

1. http server機能のみ

2. 小さい

3. コンパイル簡単

4. バグほとんどなし

・CERN httpd

1. http serverの他にProxy, Cache機能あり

2. でかい

3. コンパイルに時間がかかる。たいてい1発じゃうまくいかない。

4. バグいっぱい

...とまあ、NCSAをほめてCERNをコケにしていますが、事実、現在最も良いhttpdはUNIX上のNCSA httpdです。断言できます。<sup>\*5</sup>

CERN httpdはProxy, Cache機能があったりして機能は豊富ですが、肝心の機能が弱かったりします。<sup>\*6</sup>あとバグが多いというのがあったりするんだなあ...

ということで、httpdとしてはNCSA httpdを運用して、Cacheしたかったりする時は別口でCERN httpdを動かすのが現状ではbest choiceだと思います。

#### — note —

\*4 Mosaic Communicationsのhttpdは無視

\*5 最もこの百万石が出ている時はどうだか知らない。

\*6 例えばserver side include機能ですね。

#### 4. WWW、この混沌の世界

WWWはいつまでたってもRFCが出ない、仕様は固まらない、clientがバラバラというまさに混沌の世界です。

##### 1. clientの機能のインプリメンテーションに対し仕様が追いかけている。

HTML1では定義されていない<FORM>タグをほとんどのクライアントがサポートしているのはNCSA Mosaicがサポートしていたからに他なりません。最近Mosaic Communications社のMosaic netscapeが独自のHTML拡張をインプリメントしていくいろいろ問題になったりしているようです。

結局、現在でも<FORM>文の正式な文書はHTML2の案とNCSAの出している文しかありません。それがはたして標準になるかは誰も知りません。まさに無法地帯...

##### 2. HTTPがだめ

HTTPは現在ファイルを一つ転送してはコネクションを切るので、能率が悪い事この上ない。まあこれは現在検討中のHTTP2でmgetをサポートするよんと言っているので大丈夫であろう。

##### 3. 表現力が弱い

これはもう悲しいくらいに弱い。表さえまともに書けないので。くだらんテクを使わずにまともに割り付けができるればよいのだが...

これは将来的に改善される問題だと思うが、1.で書いたように、これに関してはいろんな奴が勝手に拡張しているので標準がない。相手の使っているクライアントを限定しないといけないというのはWWWの名が泣く。

とにかく、現状の「インプリメントしたもの勝ち」というのはどうにかしてほしい。

#### 5. 誰がWWWを見に来るのか。

例にログファイルを解析して一体どういう層が見に来るのかを見てみましょう。

これはもう1万/日のアクセスを誇るア○ムならではですね(笑)。<sup>\*7</sup>データは94/9/1から94/10/31までです。

まず国別で見てみましょう。

```
> 230423 | jp    Japan
> 117037 | unresolved
> 82256 | edu   US Educational
> 79041 | com   US Commercial
> 15888 | uk    United Kingdom
> 13074 | gov   US Government
> 10343 | ca    Canada
> 8570  | net   Network
> 5620  | au    Australia
> 5381  | se    Sweden
> 5400  | de    Germany
> 5045  | mil   US Military
> 2897  | ch    Switzerland
> 2644  | fr    France
> 2262  | org   Non-Profit Organization
> 1831  | no    Norway
> 1753  | nl    Netherlands
> 1451  | fi    Finland
> 1396  | us    United States
```

##### — note —

\*7 アコムじゃないぞ。

jpがトップなのはまあ妥当ですね。しかし逆引き登録してないところ多いですねえ。次が米国でしょうか。さすが猿の国。禁酒法を作ったのはダテじゃないですねえ(なんだそりや)。

次にドメイン別で見てみましょう。

```
> 117037 | Unresolved
> 44813 | jp.co.atom
> 11054 | jp.or.iijnet.po
> 7278 | jp.co.fujitsu
> 5918 | jp.co.sony
> 5939 | jp.ac.keio.sfc
> 5512 | com.digital.research.crl
> 5141 | jp.co.toshiba
> 4747 | jp.ac.jaist
> 4466 | jp.or.infoweb.tokyo
> 3607 | jp.co.nec.gate
> 3604 | jp.ntt.tas
> 3394 | jp.ac.uec.cc
> 3397 | jp.ac.aist-nara
> 3159 | com.att
> 3160 | jp.co.omron.ncl.ari
> 3001 | jp.or.tokai-ic
> 2889 | jp.or.rim.st
> 2812 | jp.ac.osaka-u.center
> 2735 | jp.co.mei
> 2511 | edu.stanford
> 2536 | com.netcom.ix
> 2420 | jp.ntt
> 2226 | com.sun
> 2088 | jp.or.iijnet
> 2087 | jp.ac.waseda.cfi
> 1974 | com.ti
> 1863 | jp.ac.kyoto-u.kuhp
> 1836 | jp.co.ibm.yamato
> 1782 | gov.nasa.gsfc
> 1729 | jp.co.oki
> 1669 | jp.go.aist
> 1599 | jp.co.hitachi
> 1468 | jp.go.riken
> 1455 | jp.go.jaeri.tokai
> 1510 | jp.ac.sut.kagu.cl
> 1424 | uk.co.demon
> 1414 | jp.co.toppan.tmc
> 1364 | com.hp
> 1367 | jp.ad.csi
> 1338 | com.hp.yhp
> 1381 | jp.or.win
> 1297 | jp.co.stellar
> 1289 | jp.ac.u-tokyo.t
> 1236 | ch.cern
> 1225 | jp.ac.u-tokyo.c
> 1241 | jp.co.fujixerex
> 1151 | jp.co.nikko
> 1156 | jp.co.yokogawa
> 1142 | jp.ad.iij
```

```
> 1150 | jp. kek  
> 1188 | jp. ac. kyutech. ces  
> 1066 | edu. tamu  
> 1077 | jp. ac. kyoto-u. kuis  
> 1081 | jp. co. mew  
> 1027 | edu. mit  
> 1016 | edu. ncsu. unity  
> 1009 | edu. umbc. gl  
> 1053 | jp. ac. uoeh-u. med
```

atomは内部ですからいいとして、なんとトップはiijnet.or.jpです。これからも個人のInternet userがかなりいるということがわかります。富士通とDECはrobotを運用しているようです。現在うちではrobots.txtを使ってrobotをはねる設定にしていますが、こいつらには仁義という概念がないようです。

さて、ここで同じデータを94/11/1から94/11/7までの1週間で見てみましょう。

```
> 34489 | Unresolved  
> 9119 | jp. co. atom  
> 3363 | jp. or. rim. st  
> 1951 | jp. or. iijnet. po  
> 1105 | jp. or. tokai-ic  
> 1008 | com. digital. research. crl  
> 918 | jp. co. fujitsu  
> 883 | edu. stanford  
> 831 | jp. or. iijnet  
> 817 | jp. ac. u-tokyo. c. komaba  
> 821 | jp. co. sony  
> 778 | com. sun  
> 792 | jp. ac. keio. sfc  
> 744 | com. hp. yhp  
> 703 | jp. co. sumiden  
> 649 | jp. ntt. tas  
> 654 | uk. co. demon  
> 631 | com. halcyon  
> 631 | jp. co. nec. gate  
> 610 | gov. nasa. gsfc  
> 603 | net. mcs. pr  
> 573 | jp. ac. aist-nara  
> 570 | jp. co. mei  
> 571 | jp. kek  
> 561 | edu. tamu  
> 499 | gov. nasa. arc  
> 511 | jp. ac. osaka-u. center  
> 501 | jp. co. mew  
> 515 | com. sgi
```

上位に食い込んでいるのは個人向けプロバイダーのドメインです。ここ最近の強い流れですね。tokai-ic.or.jpが多いのは公開DeleGateを上げているからでしょう。

## 6. 強引にまとめ

とにかくこれからWWWの世界はもっとぎやかになります。ブームの初期は去りました。今では一部のキレた奴がMosaic猿になるのではなく、一億総Mosaic猿時代です。

素人の新規参加も増えます。段々Internetのことはなーんにもしらないけどおいしそうだからやるとかいうデザイナーもがきが増えてきます。<sup>\*8</sup>

というわけで、本当はもっといろいろやりたかったんですが、〆切時間まであと10分なのでこのへんで...

— note —

\*8 あ、こういう時だからこそInternetのコンサルタントとかやると儲かりそう。

## X ウィンドウ (ツールキット) プログラミング pushme の製作

ういろう@MMA

僕は前からよく思っていた。なぜ「X ウィンドウ」は、「ウィンドウ」のシステムなのに、Mac や Windows のようにソフトなどをマウスでクリックして起動できないのか、なぜ MS-DOS の様なコマンドラインの「シェル」を介した起動しかできない（一部の例外を除く）のであろうか、。

そんな訳で、僕は何か「ウィンドウ」のプログラムが組みたかった。そこで、生協に行って「X ウィンドウプログラミング (ツールキット)」<sup>\*1</sup> という本を買った。その本は、「ツールキット」と呼ばれる部品化された C の関数<sup>\*2</sup> を使って、X のアプリを作るための本であった。

そんな時に、納布祭までに何かプログラムを作ろう！ というプロジェクトが我が MMA で発動した！ そこで僕は、「pushme」というプログラムを作ってみた。このプログラムは実行されるとウィンドウを作成し、与えられたファイルから、実行するための文字列を読み込んで、それを一つ一つボタンにしてウィンドウ内に表示し、それが押されると実行する。ということをするプログラムだが、これは、その「ツールキット」を使って製作されていて、たった 80 行足らずで記述されている。このように、ツールキットを使うことによって X ウィンドウのプログラムを簡単に書けてしまうのだ！

ここで、ツールキットがどんなに便利かを書いてみたい。しかし！ 僕はツールキット以外のウィンドウプログラムというものをしたことが無いのだ。これでは比較のしようがないのでとにかくツールキットの良いところと思われるところを書いてみようと思う。

まず言えるであろう事は、X でウィンドウを開くプログラムを作ろうと思ったら、ウィンドウを開くだけで膨大な量のプログラムを書く必要があると思われる。それを、ツールキットを使えばただ一つの関数を実行するだけでウィンドウを開く事ができ、そのウィンドウについてのオプション<sup>\*3</sup> 部分も一緒に判断してくれることである。

また、基本的と考えられるウィンドウの部品（例えばボタンなど）は、一つ二つの関数で書くことができる。pushme も、そのうちの一つであるコマンドウィジェット<sup>\*4</sup> を使って書かれている。これもまた、一から作ろうというときには、膨大な量のプログラムが必要となるだろう。

また、ツールキットは、標準的な部品を提供するので、ユーザーインターフェイスが統一される。そのため、ツールキットを使って作られたアプリケーションを使うユーザーは、一々新しく使い方を覚える必要が無いのである。

このように、いいことづくめのようにみえるツールキットだが、良くないこともある。簡単なところとしては、細かいところに凝ろうとするところ、できない。ということである。例えば、コマンドを選ぶボタンの形は角が丸いほうがいい、だとか、ボタンは、いつもは出っ張っているように表示されていて、押すと引っ込むようなグラフィックにしたい、などという事はできないのである。）

しかし、この様なところは、Xlib という、ツールキットよりも細かい指定の出来るライブラリを使えば実現できる。（多分簡単に、とはいかないのだろうが、、、）そもそも、ツールキットは、Xlib を使ってかれているのだ。

さて、ツールキットについていろいろ書いてきたが、僕としては、前にも書いたようにマウスでクリックするソフトが X には少ないような気がする。（もしかしたらうちの学校だけかもしれないが、、、（＾＾）そこで、ツールキットのような便利なライブラリがあるのだから、色々なツールを作るときに X だけで動くようなものを作るのならば、出来るだけマウスだけで出来るものを作りたいと思う。

\*1 実務者のための UNIX TM リテラシー・シリーズ Vol.9  
UNIX TM X ウィンドウプログラミング (ツールキット)  
HBJ 出版局 監修・斎藤 孝 [中央大学]

著者・浦新克也 安達幸司 協力・MEDACA 定価：¥2500

\*2 C 言語では、関数と呼ばれる形で、一つの一連した手続き（プログラム）を一つにまとめることができる。BASIC でいうところの GOSUB RETURN の様なものである。

\*3 X ウィンドウシステムではウィンドウを開く時には、いくつかの基本的なオプションというものが決まっていて、（例えば geometry など）大体のアプリケーションはそのオプションについてサポートしている。

\*4 例えばこのコマンドウィジェットについて考えてみると、コマンドウィジェットを定義し（ステップ 1）、そこでのコマンドが選ばれたときに呼ばれる関数を決める（ステップ 2）だけでよいのである。

# ☆ Lecture Schedule Version 1.21 ☆ for UNIX

So-Miya(miyaso-m@jed.uec.ac.jp) in MMA

平成6年 11月 7日

## 1 ▽動機▽

授業の合間に jed (J科のコンピュータ室の事) にいて時間を忘れて授業に遅刻することを未然に防ぐこと (授業をサボって jed いるためではない。はずだ、きっと...)

## 2 ▽内容▽

現在は何曜日の何時何分か、  
何時限目で何の授業なのか、または休み時間なのか、  
次に授業があるのは何時限目でまたどんな授業なのか  
何時間何分でその授業が始まるのか、  
... ということを表示するプログラムです。

MMA 調布祭出展用プログラムとして作成。ちなみに電気通信大学の時間割りをベースに作ったので、時間配分の違うところでの使用には少々のプログラム内固定データの変更が必要です。また、それぞれの時限間に休みが入ることを前提に作成されています。

## 3 ▽欠点▽

ベル音が何故か-b 指定しても鳴らないこと (Bug、ともいう(T.T。) X仕様にまだなっていないこと。(xterm の上で何とか使うという状態)

## 4 ▽必要なファイル▽

`lecschedule` プログラム本体  
`.lecturetable` 各自で作る自分の時間割ファイル

## 5 ▽実行▽

.sxsession、又は.xsession 内で、  
xterm のおいてある場所のフルパス/xterm -title 'Lecture Schedule' -geometry 80x7-0-0  
sb+-e lecsc のおいてある場所のフルパス/lecsc [options]  
例:/bin/X11/xterm -title 'Lecture Schedule' -geometry 80x7-0-0 sb+-e /home/purple/somiya/bin/lecsc -jtb  
注:.lecturetable がホームディレクトリに必要です。  
lecsc [option] [filename]

### option

- h, -? 簡易 Help
- v ヴァージョン情報を表示します。
- t 表示時間を 12 時間制にします。
- T 表示時間を 24 時間制にします。(標準)
- b 次の时限の 15, 10, 5 分前の警告時にビープ音を鳴らします。
- j 日本語モードにします。
- e 英語モードにします。(標準)

filename を指定しないときはホームディレクトリにある.lecturetable を、指定したときはそのファイルを.lecturetable の代わりに読み込みます。

例1: lecsc -t -b -j

lecsc -tbj

12 時間制、ビープ音モード、日本語モードで起動する。

例2: lecsc -j /home/miyaso-m/.lecturetable' -e

レクチャーテーブルファイル/home/miyaso-m/.lecturetable' を読み込んで、  
日本語モードで起動しようとしたけれど、英語モードに変更して起動。

相反する option を指定したとき、filename を 2 つ指定したときは後の方で指定したもの  
が有効になります。

## 6 ▽.lecturetable の書式▽

emacs 等のテキストエディターで書き込んでください。1 行 80 文字以内にしてください。

例えば、

月曜の 2 限にドイツ語 A (湯川) E 333 がある

というときは

mon2:ドイツ語A（湯川）E 3 3 3

と書いてください。

曜日は半角小文字3文字のみ受け付けます。

日曜はsun, 月曜はmon, 火曜はtue, 水曜はwed, 木曜はthu, 金曜はfri, 土曜はsatと入力してください。

続いて時限は半角文字で1から7まで受け付けます。(夜間の授業もカバー)

次に続くコロンは省略できません。

その後は半角で75文字まで、全角で37文字まで講義名などを書き込んでください。授業の無い時限は書き込まなくて結構です。また、時限、曜日の順番がばらばらでもきちんと読み込みます。

## 7 ▽プログラムについて▽

全てc言語で記述されています。Xは使ってません。(;-;) (X対応にしたらVersionを2.00に上げようと思っていたが。) treeさんに教えてもらったtermcapによる画面消去とベル音以外はたいしたことはやっていませんが、一応なにか書ないと体裁がつかないので幾つか書きます。

### 7.1 termcapについて

c言語では制御文字が使えますが、この制御文字は機種によって違います。この制御文字をプログラムの中で使い、なおかつそのプログラムをどの機種でも動くようにする、と言うのがtermcapです。UNIXでは、この制御文字が全て格納されている環境変数があり、これを参照しつつ使うことをサポートします。自分も受け売りで間違った認識をしているかもしれないが、詳しいことは各自で調べるなり、人に聞くなりしてください(a供

### 7.2 環境変数の読み込みについて

このプログラムで、標準時間割filenameは次のように決定されています。

```
char fn[256]
char *filename;
strcpy(fn, getenv("HOME"));
strcat(fn, "./lecturetable");
filename=fn;
```

このうち、getenv("HOME")が環境変数の読み込み部分で、環境変数HOMEの内容のポインタを返しています。このプログラムでは、"HOME"がなかった時の事を考慮していませんが、ないときはNULLを返すのでfilenameはルートディレクトリのlecturetable

を 差すことになるのでこのプログラムでは問題ありません。 (.lecturetableがないと  
いうエラーが出るだけです:-)。 )

### 7.3 file の読み込みについて

このプログラムの file 読み込み部分は次の通りです。(一部省略&追加)

```
static char datapool[7][7][81];
char s[81],wk[5];
FILE*fp;
if((fp=fopen(filename,"r"))==NULL) /*ファイルを開きファイルポインタを取得*/
{
    fprintf(stderr,"File %s not open\n",filename);
    exit(-1);
}
while(fgets(s,80,fp)!=NULL) /*ファイルの最後まで読み込む*/
{
    s[80]='\0'; /*文字列が 80 文字以上いった時の対策*/
    if(strlen(s)>=5) /*データが 5 文字以上*/
    {
        if(s[3]>='1' && s[3]<='7' && s[4]==':')
        {
            for(i=0;i<=2;i++) wk[i]=s[i];
            wk[3]=' '; wk[4]='\0'; /* 邪道 instr */
            n=strlen(strstr("sun mon tue wed thu fri sat ",wk))/4;
            if(n!=0) strcpy(datapool[7-n][s[3]-'0'-1],s);
        }
    }
}
fclose(fp);
```

このプログラム部分では、ファイルを開いてファイルが開けなければエラーを出し、 1 行 (80 文字) ずつ読み込み形式に合った行を選びだし、 行頭の情報に添ってデータを格納していき、 最後の行まで処理しおわったらファイルを閉じる、 という事をしています。 strlen(strstr(,)) という部分は BASIC なれした自分が作った擬似 instr です。

### 7.4 オプションの判定について

このプログラムのオプションの判定は次の通りです。(一部省略&追加)

```

void main(int argc,char *argv[])
{
    int i,j

    if(argc>1)
        for(i=1;i<argc;i++)
            if(*argv[i]!='-') /*文字列の先頭が'-'でなければ...*/
            {
                filename=argv[i]; /*それをfilenameとする*/
                break;
            }
        else /*そうでなければオプションとする*/
            for(j=i;*(argv[i]+j)!=NULL;j++)
            {
                switch(*(argv[i]+j))
                {
                    case 'h':
                    case '?': (省略:help処理) exit(0); break;
                    case 'v': (省略:version表示処理) exit(0); break;
                    case 't': flg_12=ON; break; /*12時間制*/
                    case 'T': flg_12=OFF; break; /*24時間制*/
                    case 'b': flg_bp=ON; break; /*ベル音○n*/
                    case 'j': flg_lng=1; break; /*日本語モード*/
                    case 'e': flg_lng=0; break; /*英語モード*/
                }
            }
}

```

このアルゴリズムは、全てのオプションが1文字の時に有効なものです。' -j -b -t' というように'-'を連続で指定しなくても、' -jbt' と指定すればいいようになっています。

## 8 まとめ

プログラム本体のところでは本当にたいした事をしていないので、初めてプログラムを書くという人のために初期設定の部分を説明してみた。（自分も初心者なので今までの文がまるで説明になってない気もするが...）本体部分を知りたい人はソースを解析してください。

# スーパーパーソナルの憂鬱

佃 良生 (Tsukud-Yosho) @電通大 # 9412122

平成6年 11月 8日

## 1 スーパーユーザの憂鬱

海津研でスーパーユーザをやるようになって1月たちました。今までに上げた業績は、  
DNS を引いたこと  
TeX,ghostscript,dvi2ps のインストール  
sendmail.conf の調整  
CRT の修理  
ホームディレクトリの共有、それに伴う環境の設定。

と、たくさんのお仕事を致しました。情報数理工学講座という性格上、研究室の方々は、UNIXについて詳しいわけではありません。何が憂鬱かというと、私がこれほどのお仕事をしても、評価がなかなかおいつかないのです。ユーザー教育がいかに重要かは、システムの管理をするようになって、痛いほどに感じているのですが、私は研究室の中では最年少だということもあって、私がなにか教える、または私からなにか教わるというのが非常に難しくなっています。

## 2 ユーザー教育をするか、しないか

海津研の設備を紹介します。

ワークステーション 2台 (NEWS-3460,NEWS-3470)

X 端末 3台 (JM11176 Textronix 製×2)

イーサネットでつながっているパソコン 2台 (EPSON pc-286x)

キュッパチ 2台 (pc-9801vm×2)

マッキントッシュ(Macintosh LC520)

そんな環境を、実質私一人で管理しています。

研究室の皆さんからの要望は

xterm で デリートキーが使えない。

mule で デリートキーが使えない。

マッキントッシュをネットワークにつなげ。

ワークステーションのメモリ不足をなんとかしろ。(メモリを増設するにしても、私がお金払うのか?)

どの端末も同じように使えるようにしろ。

パソコン端末の設定をしろ。

www サーバーをたち上げろ

中には横暴極まりない要求もありますが、一般のユーザーでやって欲しいことばかりです。

それ以前の問題として、研究室の方々は自分で調べようとしません。たとえば、ワークステーションのメモリ不足は、現在深刻な問題となっているのですが、発想の転換で一般ユーザーの方々に使い方を少し考えていただければ、現在の問題は問題ではないのです。

しかし、研究室の方々は、レポートや輪講でとても大変なので、UNIX の勉強などやつていられるかというのが本当のところなのでしょう。

というわけで、けなげな私としましては、誰でも使えるような道具としてのワークステーションの環境を整える方法を研究することになりそうです。

### 3 スーパーユーザとして

スーパーユーザとしての実際のお仕事は実に様々です。今までの経緯と、それを基にした課題を以下に述べます。

#### 3.1 LATEX

海津研にて、

岩本さん、LATEXって使えないんですか？

プレビューと印刷は出来ないけど、コンパイルは出来るよ。

じゃあ使えないじゃないじゃないですか。

めんどくさくってさー、インストールしなきゃって思ってるんだけどね、

今月号の UNXI マガジンに ghostscript の特集があったから僕がやってみましょう。

というわけで ghostscript をコンパイルすることになった私。はじめはテキトーにやっていました。ところが、中間発表のレポート製作に間に合うよう、即インストールしろとのお達しが下ることになります。8月下旬のことでした。

何かと不便だろうということでスーパーユーザ特権をもらうことになった私、この日はたまたま海津研と国立音大の皆様とで合コンがあったのですが、スーパーユーザになつたことがとても嬉しくて、飲み屋でもカラオケでも後悔するほど狂いまくりました。

ghostscript インストール完了！ と思いきや、虎の絵しか表示できないというファンキーな仕様であることがすぐに判明します。困ったときの MMA 頼み、砂原研の橋岡さんに教

えてもらうことにしました。メールでいろいろやり取りをしたのですが、もどかしいので砂原研から telnet をすることにします。

どうやらフォントの設定がおかしいらしい、と樋岡さん。よくよく UNIX マガジンを読んでみると、確かにそうらしいのですが、何をどうすれば.....

渡辺フォントというものがあって、ただで持ってきていいらしい。それなら、とドキュメントのとおりにフォントをインストールしました。

あのー dvi2ps もインストールしなければならないんですが.... すでに早朝 4 時だったので、おそるおそる樋岡さんに言いました。

どうしてもうまくいかない。漢字が出ない。

海津研にインストールしてあったのは、NTT jtex でした。今までやってきたことを無駄にしないためには ASCII jtex をインストールする必要があります。

結局、朝の 7 時になっても LATEX が使えるようにはなりませんでした。体力の限界を感じてその日は帰宅。

そんな日々がしばらく続くことになります。反省は、やっぱり LATEX を使う側に回ってみないことにはインストールは難しいということでした。私の場合、線形代数や、微分積分のレポートを LATEX で作成したりしましたが、数式のあるソースをプレビューするのはなかなか楽しいものでした。LATEX のインストールが完成したのはほんの 1 週間前です。それまでに何度も TeX をコンパイルしました。

### 3.2 電子メール

MMA にて

今井さん。うちの研究室からはメールが使えないんです。どうしたらいいですか。外からメールを送っても、送り返されてきちゃうんです。

なにー？ メールが使えないだと。そんなんおれだって出来るよ。sendmail.conf だろ。

えっ、でも僕初心者なので何をどうすればいいのか...

おまえ MMA だろ、合宿行って花火上げただろ、花火。

花火を上げたら、システム管理が出来るようになるんですか？

そうだよ、あったりまえじゃん。

こんな理不尽な会話に耐えることも MMA な人となってしまった今は、仕方のないことだと思います。

聞いた話によると、世界中でも sendmail.conf を調整できる人は数人しかいないということです。そんなんオレにできっこねーよ。と半分あきらめていたのですが....

きっかけは LATEX のインストールでした。わからないことがあると、おなじ MMA な人である砂原研の樋岡さんにメールで質問をするのですが、メールに関して私は教育用計算機室のアカウントを使っていたため夜中になると、連絡が取れません。

そのとき樋岡さんが砂原研にいることは知っていました。夜中の 11 時でした。LATEX のインストールに行き詰まっていた私は、なんとかして樋岡さんと連絡を取ろうと、院生

の方が持っていた I 講座共用 WS のアカウントでメールを出しました。

すいませーん、と砂原研にお邪魔をしたのですが、私が他人のアカウントを使ったことで横岡さんに本気で怒られました。

sendmail.conf の調整などとても出来そうにないと思ったので.....

CF-3.3W4.tar というファイルを頂いて、なんとかその日のうちに腐っていた sendmai.conf を正常にすることが出来ました。やっぱり西 1 号館で朝日を見ることになったのですが.....

前に一度、人がアカウントのパスワードを入力しているのを観察して、パスワードを当てたことがあるのですが、当人はむちやくちや氣を悪くしたらしく、気まずい思いをしました。

マナーの問題になるのでしょうか、他人のアカウントを使ったことに対して横岡さんに怒られたことも今になって、納得がいくようになりました。

スーパーユーザ特權の前では、一般ユーザーのプライバシーはありません。スーパーユーザとしてふさわしい人間するために、マナーを身に付け守ることも課題のひとつだと私は思っています。

### 3.3 新しい海津研ワークステーション環境と今後の保守

このたび、海津研ワークステーションの環境を一新いたしました。具体的には、どのマシンにログインしても同じように使えるように設定ファイルを各マシンどれでも問題が起こらないように書きました。これにより、どのマシンにログインしても

同じホームディレクトリにログインするようになった。

漢字が使えるようになった。

メールが読めるようになった。

ニュースが読めるようになった。

LATEX が使えるようになった。

かな漢字変換サーバを共有した。

パッと見は派手ではありませんが、各マシンの差異を吸収するためにかなりの労力を費やしているのです。

そして、変更と使い方の説明をドキュメントにして、ひと仕事終えたのですが、まだ不安定なせいもあって、研究室のユーザーの方々からの質問攻めに苦しんでいます。せっかくドキュメントを書いたのに読んでもらえません。

わたしもですが、UNIX という文化に対してはある種の誤解をしていました。パソコン少年だった私は、どうやらパソコンより高価なものにワークステーションなるものが存在するらしいという知識の下に、すべての面において、パソコンよりワークステーションの方が優れていると思っていました。

マッキントッシュや WINDOWS 流行の中で、X というユーザーインターフェースに多大な期待を抱いたことは無理もないことだと思っています。

UNIX は決してユーザーフレンドリーな文化ではなく、割と小さい単位のツールを組み合わせて、スクリプトを書くものだと教えてもらい私もそう思うようになりました。

しかし、情報数理工学講座において、道具としてワークステーションの環境を提供する身となった今、感覚的にわかりやすいユーザーインターフェースを構築することが最大の使命だと私は感じています。

具体的には、Emacs-Lisp を用いて、Mule でコントロールキーによるコマンドだけではなくメニューによるユーザーインターフェースも用意すること、マッキントッシュのファインダ、ウィンドウズのプログラムマネージャなどのようにある程度のカスタマイズがエンターユーザー自身で行えるクライアントのインストールあるいは製作、などがあげられます。

#### 4 最後に

感覚的に理解しやすいユーザーインターフェースはこれからコンピューターのあるべき姿だと思います。

つくるひと、使う人のギャップは大きくなるでしょうが、ただ、今の時点で十分大きすぎるギャップがそこにはあると私は思うのです。その差を埋める、そして使う人がなにも知らないままに安心して使える環境を用意すること。

最後に、システムの使い方についてのユーザからの質問にこころよく返答すること。その際、自分の知識のレベルを相手に強要しないこと、それこそが自分にとって難しく、そして克服すべきものなのだと私は思っています。

## 編集後記

2 E 岡田亮

okada@mma

o9310028@edu.cc.uec.ac.jp

ここにきて一年間が以前よりも早く過ぎ去っていくように感じられる。

去年の今頃も調布祭の準備をしていたわけである。そして今年も・・・

今回の百万石の出来はいかがだったでしょうか？皆さんがちゃんと締切りを守ってくれたお陰で私の編集作業は結構スムーズに進める事ができました。ただ困ったことといえば1年生の方々から『何を書けばいいんですか？』なる質問の嵐に見回されたことである。じっさいに文章を書くのはその人個人個人であって私の独断で決定するものではないのである。自分の意思でテーマを決め、それにもとづいて文章を作成すればいいのではないだろうか。結局何を書けばいいか分からなかった人も、この百万石を読んでもらえれば次回からの参考となるのではないだろうか。

もしこの百万石に対する感想（本文・編集）などございましたら、私のところやMMAまでE-mailなどで送ってください。また次回何かの機会でお目にかかるのを楽しむにしております。（1994年11月岡田 亮）

---

## 百萬石

発行日 1994年11月18日

発行人 村上敏之

編集人 岡田亮

Special Thanks MMAの皆さん

発行所 電気通信大学MMA

Copyright(C) 1994 MMA

本誌からの無断転載を禁じます。