

経済学部電子メールサーバの利用状況

今 井 慈 郎
三 谷 宗 子
本 田 道 夫

I はじめに

香川大学の国内インターネットへの接続は、岡山大学工学部情報工学科、松山研究室¹⁾と香川大学の情報処理センターとの間でワークステーション同士のUUCP接続で始まった。当時、国内ではインターネットという表現自身も一般的ではなく、JUNET²⁾(Japan UNIX NETwork)と呼ばれていた。その後、平成5年11月に64 Kbpsの専用回線によるインターネットへのIP接続をはたしている。すなわち、文部省学術情報センターのインターネットバックボーンSINETが提供するIPルーティングサービスへの参加である。情報処理センターに導入されたルータと、神戸大学内に設置されたSINET集合ルータとの間でのIP接続が実現され、学内LANとインターネットとが相互に接続できる環境が構築された。学内LAN(香川大学キャンパス情報ネットワーク)は、平成5年度の補正予算、約1億5千万円による100 Mbpsの光ファイバーループ幹線、イーサネットによる支線LANの敷設およびファイルサーバやメールサーバなど各種UNIXマシンの導入でスタートし、平成7年度から稼動し始めた情報処理センター新システムで一応の完成をみた。

ネットワーク環境における最も基本的なサービスの1つとして電子メールサ

E-mail: {imai, mitani, honda} @ec.kagawa-u.ac.jp

1) 松山先生は現在は京都大学教授に転出。

2) JUNETではUUCP接続とIP接続との混成でネットワークが組織されており、現在のインターネットがIP接続を主力とするのとは大きな違いがあった。

ービスがあげられる。UUCP 接続であれ、IP 接続であれ電子メール自体は実現可能なサービスである。しかし、それまで UUCP 接続による 2 時間に 1 回のモデムを介した情報交換という運用形式であったサービス状況はインターネットへの IP 接続を果すことで大幅に改善された。インターネットへ IP 接続される以前も学内のマシン同士は IP 接続されていたため、学内の研究者同士が IP 接続の特徴を活かした電子メールサービスを利用でき、高速な情報交換も可能であった。一方、インターネットとの IP 接続が実現された後の電子メールサービス環境では、学内外の境がなくなり、国内の大学同士はもとより国外の大学や研究機関との間で、数十秒から数十分のリアルタイムなメールの相互交換が可能となった。情報交換の手段としては電話や FAX 以上の利便性が享受できる情報通信サービスの形態と言える。

電子メールサービスは学外との高速なメール交換と共に、同一マシン内に登録されているユーザ同士での非同期な意見交換や情報伝達も容易に実現できる点が有用である。一方、メールサーバは UNIX マシンであるため、メールサーバに登録されているユーザであれば、telnet や ftp などのコマンドを用いて遠隔地に存在する計算機環境を利用することもできる。インターネットに接続している UNIX マシンでは多様なコンピューティングサービスが可能である。本稿では電子メールサービスを中心に、インターネットへの IP 接続のいきさつ、メールサーバの学部移管や運用体制、メールサーバである UNIX マシンの利用状況などについて述べている。また、メールサーバを運用してきたことでどのような問題が発生したかについても具体的に説明している。

II インターネット接続に伴う情報サーバの分散管理

SINET インターネットバックボーンへの IP 接続は、当初の思惑を越えて英断とも呼べる結果となった。平成 5 年当時、香川大学の所属する第 6 地区の計算機センター連絡会議（大阪大学大型計算機センターが主管）では、技術力や信頼性に関して SINET への批判が相次ぎ、自己資金が叶えば WIDE に代表される研究者・技術者の揃ったインターネットプロジェクトへの IP 接続が当然とみなされる雰囲気であった。『盆暮れに止まる SINET に接続するのは理解に

苦しむ』などの発言も相次いでいた。大阪大学を中心に ORIONS と呼ばれる学術関係機関の地域ネットワークや関西の民間企業を糾合する形で構成された WINC 地域ネットワークなどが産声をあげていた時期でもあり、これらは今でもネットワークの運用資金（の一部）を構成員である各組織が持ち寄る形態で全体が運営されている。

香川大学の場合、最寄りのインターネット接続ノードまでの専用回線のレンタル経費だけでも月額十万以上であったため、SINET 以外でインターネットへ接続するには、経費負担が問題となり、学内のコンセンサスが得難いという状況にあった。しかし、ORIONS に所属する大学からは『なんだ香川大学は SINET に繋ぐのか』的な評価を受けたという印象があった。当時最も近い SINET ノードが神戸大学にあったことから、結局 SINET への接続には、高松一神戸間に STNet-JT-OMP などの電力 JR 系のネットワーク業者が提供する 64 Kbps の専用回線をレンタルすることとなった³⁾

技術力や信頼性が低いなどと酷評された SINET であったが、IP 接続当時から学術情報センターのサポートは予想を裏切る(?)ほどの好意的で高度な技術力を持った対応であった。ネットワーク係の内田邦夫氏には一貫して的確な助言などをいただき、香川大学の学内 LAN を運用する上で不可欠な貢献をいただいている。適切かつ迅速な指示⁴⁾に今井などは知識も技術も追従できないまま、『それでお願ひします』を連発する状態であった。当時、情報処理センターを担当していた KNES の武藤直美氏が初代ネームサーバ(kagawa-ns)を設定したり、内田氏がリモートでルータ等の設定を行うなど、香川大学がインターネットへ IP 接続を果たしたのは他力本願であったことは筆者達の記憶にも新しい。着任後、半年程の今井にとって初めての経験ばかりでまさに『何がどうなっているのかチンプンカンプン』であり、専任教官とは名ばかりの状態であった。しかし、IP 接続と同時に解決すべき重要な問題も多かった。電子メールサービス体制の再構築もその 1 つであった。

3) 現在は岡山大学、香川大学間に Stnet-JT-Tinet の同じく電力 JR 系のネットワーク業者の提供する 1.5 Mbps の専用回線をレンタルしている。

4) 内田氏は結構せっかちな性格とお見受けするが…。

それまで、言わば細々と UUCP 接続で行われていた電子メールサービスは、

- 利用者の絶対数が少ない
- 全学で1つの UNIX マシンをメールホストとして利用する
- 学内 LAN を介してリモートアクセスし、メールホスト上で読み書きするという状況であった。このようにメールホストを中心に据えた集中型の電子メールサービス体制の利点として、

- 1) マシン管理が容易
- 2) システム設定が簡単

などが挙げられた。結果として、マシン導入業者に管理・設定などの依頼も比較的気安く行うことが可能であった。しかし、ユーザへのサービスという観点から見ると、

- 1) 同一マシンに多数がログインすることでレスポンスが低下
- 2) 登録ユーザ名の衝突を回避するため実名より記号的な登録名（課題番号）を採用
- 3) 集中管理方式によりシステムダウン時の影響が深刻
- 4) 全学的コンセンサスを前提としているため、学部レベルの配慮や企画が困難

- 5) 学生など多数ユーザを前提とできないシステム構成上の制限

など問題点も多かった。現在と比較すればユーザ数も比較にならないほど少なかったので問題が顕在化しなかったが、このままの体制でインターネットへの接続方式が UUCP ベースから IP ベースへと移行すれば当然メールサービスの問題点がいきなり表面化してしまう危険性があった。インターネットへの IP 接続を果たした情報処理センターにとって、電子メールサービス体制の再構築が当面する重要課題と位置付けられた。

一方、UUCP 接続当時から、香川大学の電子メールサービスには拡張性を指向した配慮がなされていた。メールアドレスのクラス分けである。単一のメールホスト (kugw) であっても、情報処理センターのユーザなら “cc.kagawa-u.ac.jp” を、経済学部のユーザなら “ec.kagawa-u.ac.jp” をそれぞれメールサイトとするように別名機能 (mail alias) を利用して、仮想的に複数のメールサ

イトが存在するよう見せ掛ける設定を行っていた。このような設定には、遠くない将来、各部局毎にメールサーバを割り当てることで、複数のメールサーバを分散配置する方式を実現する意図があったと思われる。多くの場合、メールサイトはメールサーバに一对一に対応している。そこで、IP 接続後に情報処理センターが提供すべき新しい電子メールサービスの基本方針として、

- 1) 各学部 to メールサーバを設置し、分散処理方式で電子メールサービスを実現
- 2) メールサーバの管理を各々の部局に可能な限り一任
- 3) 個々のメールサーバ管理者への権限委譲の促進
- 4) ユーザ登録などを部局内にクローズさせ、ユーザ名の衝突を低減
- 5) 学部独自の運用形態も可能な限り承認

などを仕様とするよう対策が図られた。サービスの質的レベルを部局によらず一定に維持するのは情報処理センターの責務であるとしながらも、分散管理体制には結果として均一サービスではなくなる可能性も内在させている。これは電子メールサービスに関する『小さな政府、可能な限りの地方分権』の促進を意味し、学内ネットワークにおける典型的な分散処理の事例となった。

UUCP 接続当時は情報処理環境を統一的に提供するという意図から個々のユーザを課題番号で登録していた。そのため、見掛け上電子メールのアドレス (E-mail アドレス) に課題番号ではなく個人名を許容できるよう別名機能を用いていた。しかし、別名機能を用いて見掛け上のみ個人名の E-mail アドレスを使用しても、メール送信時には先方の受信者にとって送信者を判読しづらい課題番号による送信者名が付記されてしまう設定は回避できなかった。このように他組織には馴染まない現象が発生するため、個人名でのユーザ登録を採用しなければ回避できそうもない事態と判断された。課題番号でのユーザ登録が基本的に、CPU/ファイルサービスを対象とした情報処理環境中心の方針であったのに対して、メールサービスのような情報通信環境においては、ユーザ登録を個人名で行うという方針が避けられない状態であった。このような事実 1 つをとってみても、インターネット時代の情報処理センターにとって、情報処理環境の効率的な提供ばかりでなく、情報通信環境の効果的な提供が重要な業務

である証左となっており、まさに電子メールは象徴的な事例となっていた。

個人名でのユーザ登録の場合、扱う範囲が大きくなればなるほど同姓が存在する発生確率も上昇する。部局毎に閉じたユーザ登録が可能ならば、ユーザ登録時の同姓者同士のユーザ名をどうネーミングするかという調停も、全学レベルで行うよりはるかに円滑に実現できる。各部局のメールサーバ管理者にシステム管理やユーザ登録などの業務を依頼する以上、権限の委譲は表裏一体となる。学部内のニーズに応じて学生・大学院生のユーザ登録を試験的に行うなどのサーバ毎の運用方針を全学レベルで議論する必要がなくなった点も大きな特徴である。まさに、小回りの効く迅速な管理運用が可能となった。

しかし、新たな問題も同時に生じてきた。それまでも情報処理センターの業務は業績評価にならないなどの理由で、必ずしも広く学内に協力者を得るのが難しい状況であった。このため、各学部からメールサーバ管理者を推薦してもらうことの意義などが正確には理解されておらず、スタート当時から『分散処理に応じた分散管理』が順調に実施できた訳ではなかった。その間、NEC (SNES および KNES) の人的貢献は特筆すべきであったと思われる。また、ユーザ登録など比較的初等レベルのシステム管理から始まり、徐々にメールサーバを主体的に管理運用することができるようになったのは、各学部のメールサーバ管理者の尽力によるところが大であった点も特筆すべきである。我が経済学部の専用メールサーバ(ホストの UNIX マシンのニックネームは *fourier*)の導入もこのような全学的状況下にあつて、情報処理センターから独立する形で実現したことは言及するまでもない。しかし、経済学部においては、兼任教官が欠員状態であり、学部への連絡周知が十分でなかった点などは今井の認識不足などが主な原因であった。明記してお詫びしたい。

III メールサーバ *fourier* の設定と運用方針

経済学部へ管理運用が一任されているメールサーバは日本電気製 UNIX ワークステーション (EWS 4800/320) であり、*fourier* と呼ばれる。マシン仕様は

- CPU : R4000 (40 MHz)
- 主記憶 : 32MB

- ハードディスク：1.1GB
- 無停電電源装置

となっている。この分野の進歩は急激で、EWS4800/320の後継機であるEWS4800/320VXにはCPUとして、R4400SC(100 MHz)が搭載されている。導入時に最新機種であったと言えども2年の経過で既に相対的機能低下が感じられるのも故無しとしない。従って、近い将来新機種を導入する必要も生じるであろう。その意味でもソフトウェアの設定方針や運用方針を文書化しておくことは有効となる。

III-1 fourier のシステム設定

IP 接続によってインターネット上の様々な計算機とリアルタイムに情報交換できるようになるが、アクセスしたい対象となる計算機の IP アドレスを取得することが重要である。ネームサーバは計算機のホスト名と IP アドレスとを対にしてデータベース化して管理しており、個々の UNIX マシンからの照会に対して当該計算機のホスト名から IP アドレスを答えるアドレス変換サービスを提供する。fourier など個々の UNIX マシンにはリゾルバ(resolver)と呼ばれるルーチンが用意され、ユーザが世界中の計算機へアクセスする時にはネームサーバと交信して IP アドレスを取得する[2]。リゾルバが参照するネームサーバの IP アドレスはファイル“/etc/resolv.conf”に記述する必要がある。香川大学のネームサーバが変更される度に fourier のリゾルバの設定も変更される。当初は初代一次ネームサーバ kagawa-ns.nw.kagawa-u.ac.jp (133.92.6.17)を参照するよう初期設定していた。その後、一次ネームサーバの不調を改善するために別途用意した二次ネームサーバ⁵⁾を参照するよう一時的に変更し、現在では二代目の一次ネームサーバである ns1.kagawa-u.ac.jp (IP アドレスは初代と同じ 133.92.6.17) を立ち上げたため、リゾルバの設定を再変更している。ネームサーバを参照することで、telnet/ftp などの利用時に

5) 二次ネームサーバ (IP アドレス：133.92.6.18) は当初 kagawa-ns2.nw.kagawa-u.ac.jp と呼ばれていたが、情報処理センター新システムが導入されたのを契機にマシンを更新し、新たに ns2.kagawa-u.ac.jp と呼ぶよう変更した。

正規のドメイン名 (Fully Qualified Domain Name, FQDN) を用いて、リモートアクセスしたい計算機を指名することができる。

リゾルバを正しく設定することは、電子メールサービスを実現する上でも重要な意味を持っている。UUCP 接続時の香川大学のメールサービスはメールホストとなる特定のメールサーバに総てのメールを集配送させ、メールの転送はもとより読み書きなどの処理までも集中させるメールホスト一局集中型であった。一方、IP 接続時のメールサービスでは、

1) 電子メール送信時には

- リゾルバを用いて宛先のホスト名をネームサーバに問い合わせ、
- IP アドレスを取得し、
- 先方が稼働していることを確認して直接メールを送信する。

2) 電子メール受信時には、

- 送信メールサーバからのアクセスが正しくこちら宛であることを確認し、
- メール関係の aliases などの処理した後、
- 受信者となるユーザが存在することを確認して、
- メールを受信してユーザ名の私書箱とも言えるメールプール領域に蓄える。

などの処理を効率良く行う必要がある。このようなメールサービスを実現するソフトウェアは、“MX 対応の sendmail(/usr/ucblib/sendmail.mx)” と呼ばれるメール送受信・管理デーモンである。MX 対応とはネームサーバが提供する MX (Mail eXchange) レコード情報を参照して IP ダイレクトにメールを送信する機能である。また、デーモンとは UNIX で伝統的に使用されるバックグラウンドで実行可能なサービスプログラムのことである。しかも、sendmail は SMTP (Simple Mail Transfer Protocol) に対応してあらゆる UNIX マシン上の sendmail とメール交換が可能であり、会話型処理やデバック、あるいは設定チェックなど、メールサービスを実現する上でほとんどの機能を提供してくれる。

このように機能の高い sendmail であるが、問題点もある。その 1 つは sendmail の動作を規定している“設定ファイル sendmail.cf (/var/ucblib/send-

mail.cf) が通常の管理者にとって容易に記述できないばかりか、判読も容易でない点である。従来、sendmail.cf はゼロから記述するのではなく、類似の設定を見つけ出し、それを模倣するのが最善と言われている。fourier 上の sendmail.cf もこのような趣旨に沿って、独自に sendmail.cf を記述する方法をとらず、既存の sendmail.cf を一部変更して再利用する方法を選択している。sendmail.cf の設定仕様は以下の通りである。

1) “ユーザ名@ec.kagawa-u.ac.jp” 宛のメール受信

これは言わば正式の E-mail アドレスであり、ネームサーバにもメールサイト“ec.kagawa-u.ac.jp” ならば受信するマシンは“fourier.ec.kagawa-u.ac.jp” であると設定されている。従って、この場合もユーザ名の記述が正しければ fourier でメールが受信されなければならない。

2) “ユーザ名@fourier.ec.kagawa-u.ac.jp” 宛のメール受信

これは FQDN として記述されたホスト名であり、指定されたユーザ名に対応するユーザが存在すればメール受信は必ず成功させなければならない。

3) 学内メールサーバへのメール送信

ネームサーバ参照を行わないでも学内のメールサーバには正確にメールを送信できる方が信頼性が高い。仮にネームサーバが落ちていても、学内のメールサービスは稼働できる。

4) MX レコードを有するメールサーバへのメール送信

学外のメールサイトへメールを送信する場合にはネームサーバを参照して MX レコードあるいは A レコードを参照し、相手先の IP アドレスを取得してリアルタイムでメールを送信できる。ネームサーバへのアクセスをスムーズに行うためリゾルバを適切に設定する必要がある。

導入当時の fourier の sendmail.cf も、当時の設定仕様に適応できるように初期設定されていた。しかし、情報処理センターで行ったネームサーバの再設定や全学的メールサービス体制の見直しなどを受けて、fourier の sendmail.cf も少しずつ修正されている。とは言うものの、基本的にネームサーバ参照を前提としたメール送信設定への移行や、経済学部講師、加藤大志朗先生による“ec.kagawa-u.ac.jp” のドメイン配下に試験的サブドメインの設定などはいずれ

も一般ユーザにとってほとんど変化とは感じられない修正であった⁶⁾

パソコンユーザにとって、UNIX マシン上のメールサービスを初め UNIX ユーザインターフェースのみで利用しなければならず、『繁雑で利便性がない』という指摘の声が大きかったのも事実であった。曰く『Emacs を利用したメールの読み書きは不便である』、『パソコンから UNIX に telnet 接続すること自体が面倒である』、『日本語コードが UNIX とパソコンとでは異なる』等々。このような批評に対して、

- 1) 10 名以上のユーザが telnet でログインして、Emacs (Rmail モード) でメールの読み書きを行うのは、fourier のハードウェア仕様からも適切な応答時間を期待することができない。応答時間の短縮やログイン不能状況の回避を図らなければならない。
- 2) メールユーザの多数がパソコンユーザである以上、メールサーバである fourier はパソコンユーザにとっても利用し易い環境を提供しなければならない。
- 3) UNIX マシン上に大きなサイズのメールデータを少しでも残さないよう、パソコン上までメールを転送できる仕組みを導入し、UNIX 上のメールユーザと共存可能な環境設定が必要である。

などの観点から、本田によって fourier への popper 導入が決断された。

popper 自体は sendmail などの大型のデーモンソフトウェアとは異なり、インストールも簡単である。パソコン (Macintosh や WindowsPC) などでは POP3 (Post Office Protocol) と呼ばれるメール転送簡易プロトコルに対応したメール読み書きソフトウェア (mailer/mail reader) が急速に普及し始めていた。popper は UNIX 上で稼働し、POP3 手順でメールデータをパソコンに転送する機能を有する。popper を fourier にインストールする作業は当時教育学部大学院生であった光家功人氏⁷⁾に依頼された。多くのパソコンユーザが比較的

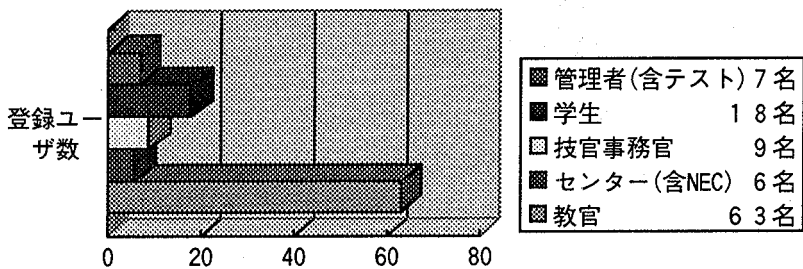
6) 最終版の sendmail.cf を本稿に掲載すべきであるが、サイズの大きな点、一般ユーザにとって意味不明(?)な記述である点などの理由で躊躇された。fourier ユーザであれば telnet でログインし、% more /var/ucplib/sendmail.cf など確認できる。

7) 当時、情報処理センターのアルバイト学生であった。

簡単に電子メールサービスを利用できるようになった現在, popper のインストールは不可欠であったことは容易に理解される。popper 自体がフリーウェアであり, 当時日本電気がフリーウェアをサポートしないという方針であったため, 情報処理センターが正式サポートをするか否かの議論⁸⁾を行っていた最中であったことを考えると, まさにユーザの利便性を優先させた英断であったと思われる。今や法学部, 事務部あるいは学生用のメールサーバに至るまで標準的にインストールしておくべき代表的なフリーウェアと言える。

III-2 ユーザ登録状況

fourier に登録されているユーザは, 経済学部の教官を中心に, 情報処理センター関係者, 事務系職員, 学生などから構成されている。図III-1 は平成7年



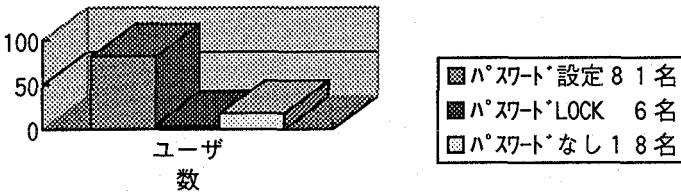
図III-1 fourier に登録されているユーザの分類

12月1日現在の登録者分布を示している。試験的に登録していた学生ユーザは基本的に学生用メールサーバに移行させる予定であり, 事務系職員の多くも事務用メールサーバに登録中である。一方, 情報処理センター関係者や管理者グループは今後とも fourier 上に留まって運用支援に寄与することになっている。

経済学部の場合, 総ての教官に可能な限り同等な権限を準備する制度が望ま

8) 本田は情報処理センター長として popper のインストールに積極的であったが, ソフトウェアへの知識がないなどを理由に今井は消極的であった。

しいという見解が支配的である。UNIX マシンではこのような見解がいつも妥当であるとは言い難いが、fourier の運用を開始した平成6年度当初では総ての教官をユーザ登録することとした。ユーザとして登録することを総ての教官から要請された訳ではないので、言わば勝手な判断であり、後にいくつかの問題点を生じさせた。登録された個々のユーザが主体的にパスワードを設定し、UNIX マシンの機能の1つとしてメールサービスを利用すると想定していたが、平成7年12月1日現在では図III-2のようなパスワード設定状況にある。パスワードを設定していないユーザに設定を強く要請すべきか、ユーザ登録を解消すべきか管理者としては大いに迷っており、解決すべき問題の1つである。



図III-2 fourier ユーザのパスワード設定状況

security hole という概念の徹底が今後の課題となろう。ちなみに、パスワードを LOCK しているユーザは基本的に本人からの要請に応じた措置である。

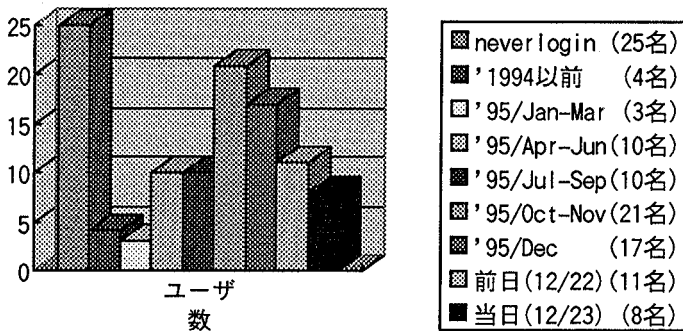
IV メールサーバ fourier の利用状況

本節では fourier の利用状況を具体的に示すため、登録されたユーザがどの程度の頻度でアクセスしているかを定量的に調査し、ログインの状況、平均アクセスの状況、パソコンの電子メールリーダの利用状況など perl スクリプトを用いて統計処理して結果をグラフ化している。また、これまで fourier を運用してきた中で発生した問題点についても具体的に考察している。

IV-1 利用状況の視覚化

UNIX にはシステム管理を円滑に行うため、いくつかのログ (記録) がとられている。登録されている総てのユーザが最後に login した日時なども記録さ

れている。一方、fourier には lastlog と呼ばれるフリーウェアをインストールしており、この最終ログイン日時を分析してユーザの利用動向を確認することができる。本稿をまとめる上で平成7年12月23日時点の lastlog を調べたところ、図IV-1のような結果を得た。これからいくつかのことが推測される。fourier は電子メールサービスを主体としている UNIX マシンであるが、ほぼ毎日10名から20名程度のユーザが telnet/ftp コマンドでアクセスしている可能性がある。一度もログインしていないユーザもかなりの数にのぼっている。UNIX 自身が記録しているユーザログ情報は伝統的なアクセスコマンドである telnet/ftp でのログインを対象としている。従って、telnet/ftp を利用したユーザがどのような頻度で fourier を利用しているかをもう少し詳細に調べて利用動向をまとめることが可能である。一方、メールサービスをパソコンから利用しているユーザの利用動向の調査は必ずしも容易ではないので工夫が必要である。



図IV-1 telnet/ftp での利用状況の集計 (平成7年12月23日現在)

まず、telnet/ftp を利用しているユーザ数がどの程度か、1日に平均して何回 fourier にアクセスしているか、telnet と ftp とのアクセス比率はどうか、などをサンプリングしたログを用いて解析し、fourier ユーザの利用動向を定量化してみる。利用動向の調査は平成7年10月20日から12月22日までの2ヶ月間に行い、週の前半として火曜日の、後半として金曜日のアクセス状況をサンプリングしている。このサンプリングとデータの収集には UNIX のシステム管理

に広く利用されている cron を用いた。また、収集したデータの処理には汎用スクリプト言語の perl [3] を利用している。以下にデータ処理で用いた perl スクリプトを示す。

```
#!/usr/local/bin/perl
open (fh, "-");
while ( $ _ = <fh> ) {
    s/-//g;
    s/¥s+// /g;
    s/still logged in/Now/g;
    @buf = split;
    $output = join ( ' ', @buf[0], @buf[1], @buf[4], @buf[5], @buf[6], @buf
[7] );
    unless (/wtmpx/) {
#extract the string such as "/var/adm/wtmpx begins..."
        print $output, "¥n";
    }
}
}
```

図IV-2 perl スクリプト last1.pl

紹介するスクリプトは3つの部分、すなわち、last1.pl, last2.pl および last3.pl から構成されている。last1.pl はログデータの整形を、last2.pl は利用時間の集計を、last3.pl は測定日ごとの telnet アクセス数, ftp アクセス数, 利用ユーザ数を算出するのに利用している。

```
#!/usr/local/bin/perl
while ( $ _ = <STDIN> ) {
    s:/ /g;
    @buf = split;
    unless ( @buf ) { # nothing for blank line
        print $ _;
        next;
    }
}
```

```

$MonthDay = join('/', @buf[2], @buf[3]);
$fromTime = join(':', @buf[4], @buf[5]);
if ( @buf[6] eq "Now" ) {
    $useTime = 3;
    $minute = @buf[5] + $useTime;
    if ( $minute > 60 ) {
        $hour = @buf[4] + 1;
        $minute = $minute - 60;
    } else {
        $toTime = join(':', $hour, $minute);
    }
} else {
    $toTime = join(':', @buf[4], $minute);
}
if ( (@buf[4] == @buf[6]) && (@buf[5] == @buf[7]) ) {
    $useTime = 1;
} else {
    $useTime = (@buf[6] - @buf[4]) * 60 + (@buf[7] - @buf[5]);
}
$toTime = join(':', @buf[6], @buf[7]);
}
if ( @buf[1] ne "ftp" ) {
    $UserAccess = join(' ', @buf[0], "telnet");
} else {
    $UserAccess = join(' ', @buf[0], @buf[1]);
}
$output = join(' ', $MonthDay, $fromTime, $toTime, $useTime, $UserAccess);
print $output, "\n";
}

```

図IV-3 perl スクリプト last2.pl

それぞれのスクリプトは、実行可能ファイルであり、個別にフィルタとして利用できるよう設計されているので、収集されたログデータから必要な情報を抽出するのに組み合わせて利用することも可能である。

```

#!/usr/local/bin/perl
sub initialize {
    $telnetCnt = 0;
    $ftpCnt = 0;
    $userCnt = 0;
    open (TempFile, ' | sort | uniq>temp.temp');
}
sub outputResult {
    close (TempFile);
    open (fh, "temp.temp");
    while ( $_ = <fh> ) {
        ++$userCnt;
    }
    $output = join(' ', $MonthDay, $telnetCnt, $ftpCnt, $userCnt);
    print $output, "¥n";
    $done = 1;
}

&initialize();
while ( <stdin> ) {
    @buf = split;
    if ( @buf ) {
        if (@buf[5] eq "telnet") {
            ++$telnetCnt;
        }
        if (@buf[5] eq "ftp") {
            ++$ftpCnt;
        }
        $MonthDay = @buf[0] if ( $Month ne @buf[0] );
        print TempFile "@buf[4]¥n";
        $done = 0;
    } else {
        &outputResult();
        &initialize();
    }
}
}

```



```
&outputResult() unless $done;
```

```
IV-4 perl スクリプト last3.pl
```

これらのスクリプトを実際に動作させて、cron を用いて収集したログデータ
“last. fourier” を処理する手順を以下に示す。

```
(fourier) imai[1] last
```

```
imai pts/4 e0089 Fri Jan 12 18:22 still logged in
ando pts/4 e0017 Fri Jan 12 17:37 - 17:37 (00:00)
yanagi pts/4 133.92.90.21 Fri Jan 12 17:12 - 17:12 (00:00)
yamada pts/7 133.92.62.24 Fri Jan 12 16:57 - 17:06 (00:08)
ozaki pts/4 133.92.94.149 Fri Jan 12 16:57 - 16:57 (00:00)
yamada pts/4 133.92.62.24 Fri Jan 12 16:55 - 16:56 (00:01)
tominaga pts/5 e2011 Fri Jan 12 16:49 - 17:48 (00:59)
yamada pts/4 133.92.62.24 Fri Jan 12 16:49 - 16:51 (00:02)
yanagi pts/6 133.92.90.21 Fri Jan 12 16:42 - 16:48 (00:05)
inoues pts/6 e0029 Fri Jan 12 16:42 - 16:42 (00:00)
yanagi pts/6 133.92.90.21 Fri Jan 12 16:40 - 16:40 (00:00)
ando pts/7 e0017 Fri Jan 12 16:38 - 16:40 (00:01)
mitani pts/7 e0018 Fri Jan 12 16:37 - 16:37 (00:00)
```

```
<<中略>>
```

```
inoues pts/5 e0029 Fri Jan 12 09:44 - 09:44 (00:00)
ando pts/5 e0017 Fri Jan 12 09:39 - 09:40 (00:01)
honda pts/4 e0001 Fri Jan 12 09:26 - 09:51 (00:24)
inoues pts/5 e0029 Fri Jan 12 09:24 - 09:29 (00:04)
mitani pts/4 e0018 Fri Jan 12 09:04 - 09:24 (00:20)
yokoyama pts/3 e0043 Fri Jan 12 08:57 - 18:54 (09:57)
ueda pts/3 133.92.8.82 Fri Jan 12 08:50 - 08:56 (00:06)
inoues pts/3 e0029 Fri Jan 12 08:48 - 08:49 (00:00)
shishido ftp e3021 Fri Jan 12 07:47 - 07:47 (00:00)
shishido ftp e3021 Fri Jan 12 07:38 - 07:38 (00:00)
shishido ftp e3021 Fri Jan 12 07:24 - 07:24 (00:00)
```

```
/var/adm/wtmpx begins Fri Jan 12 07:24
```

```
(fourier) imai[2] cat last. fourier
```

ozaki	pts/5	133.92.94.148	Fri Oct 20 16:47 - 16:47 (00:00)
mitani	pts/4	e0018	Fri Oct 20 16:47 - 16:53 (00:05)
mitani	ftp	e0018	Fri Oct 20 16:46 still logged in
imai	pts/3	e0089	Fri Oct 20 16:46 still logged in
ueda	pts/3	133.92.8.82	Fri Oct 20 16:43 - 16:44 (00:01)
imai	pts/4	e0089	Fri Oct 20 16:30 - 16:30 (00:00)
mitani	pts/3	e0018	Fri Oct 20 16:26 - 16:32 (00:05)
mitani	ftp	e0018	Fri Oct 20 16:26 - 16:32 (00:06)
mitani	ftp	e0018	Fri Oct 20 15:43 - 15:54 (00:11)
mitani	pts/3	e0018	Fri Oct 20 15:23 - 16:14 (00:51)
mitani	ftp	e0018	Fri Oct 20 15:23 - 15:38 (00:15)
mitani	pts/3	e0018	Fri Oct 20 14:46 - 14:47 (00:01)
s90e740	pts/3	e0028	Fri Oct 20 14:43 - 14:43 (00:00)
s92e748	pts/3	gauss2	Fri Oct 20 14:38 - 14:39 (00:01)
mitani	pts/3	e0018	Fri Oct 20 14:27 - 14:30 (00:02)
s92e702	pts/5	e0028	Fri Oct 20 13:49 - 14:01 (00:11)
imai	pts/3	e0089	Fri Oct 20 13:31 - 13:55 (00:23)
hisamatu	pts/3	CSC002	Fri Oct 20 13:00 - 13:01 (00:00)
hisamatu	pts/3	CSC002	Fri Oct 20 12:54 - 12:57 (00:02)
hisamatu	pts/3	CSC002	Fri Oct 20 12:45 - 12:47 (00:02)
honda	pts/3	e0001	Fri Oct 20 12:38 - 12:40 (00:01)

<<以下, 省略>>

(fourier) imai[3] cat last fourier | last 1. pl | more

ozaki pts/5 Oct 20 16:47 16:47

mitani pts/4 Oct 20 16:47 16:53

mitani ftp Oct 20 16:46 Now

imai pts/3 Oct 20 16:46 Now

ueda pts/3 Oct 20 16:43 16:44

imai pts/4 Oct 20 16:30 16:30

mitani pts/3 Oct 20 16:26 16:32

mitani ftp Oct 20 16:26 16:32

mitani ftp Oct 20 15:43 15:54

mitani pts/3 Oct 20 15:23 16:14

mitani ftp Oct 20 15:23 15:38

mitani pts/3 Oct 20 14:46 14:47

s90e740 pts/3 Oct 20 14:43 14:43

s92e748 pts/3 Oct 20 14:38 14:39
mitani pts/3 Oct 20 14:27 14:30
s92e702 pts/5 Oct 20 13:49 14:01
imai pts/3 Oct 20 13:31 13:55
hisamatu pts/3 Oct 20 13:00 13:01
hisamatu pts/3 Oct 20 12:54 12:57
hisamatu pts/3 Oct 20 12:45 12:47
honda pts/3 Oct 20 12:38 12:40

<<以下, 省略>>

(fourier) imai[4] cat last fourier | last 1. pl | last 2. pl | more
Oct/20 16:47 16:47 1 ozaki telnet
Oct/20 16:47 16:53 6 mitani telnet
Oct/20 16:46 16:49 3 mitani ftp
Oct/20 16:46 16:49 3 imai telnet
Oct/20 16:43 16:44 1 ueda telnet
Oct/20 16:30 16:30 1 imai telnet
Oct/20 16:26 16:32 6 mitani telnet
Oct/20 16:26 16:32 6 mitani ftp
Oct/20 15:43 15:54 11 mitani ftp
Oct/20 15:23 16:14 51 mitani telnet
Oct/20 15:23 15:38 15 mitani ftp
Oct/20 14:46 14:47 1 mitani telnet
Oct/20 14:43 14:43 1 s90e740 telnet
Oct/20 14:38 14:39 1 s92e748 telnet
Oct/20 14:27 14:30 3 mitani telnet
Oct/20 13:49 14:01 12 s92e702 telnet
Oct/20 13:31 13:55 24 imai telnet
Oct/20 13:00 13:01 1 hisamatu telnet
Oct/20 12:54 12:57 3 hisamatu telnet
Oct/20 12:45 12:47 2 hisamatu telnet
Oct/20 12:38 12:40 2 honda telnet

<<以下, 省略>>

(fourier) imai[5] cat last fourier | last 1. pl | last2. pl | last3. pl
Oct/20 31 4 14
Oct/24 39 6 16
Oct/27 54 4 17

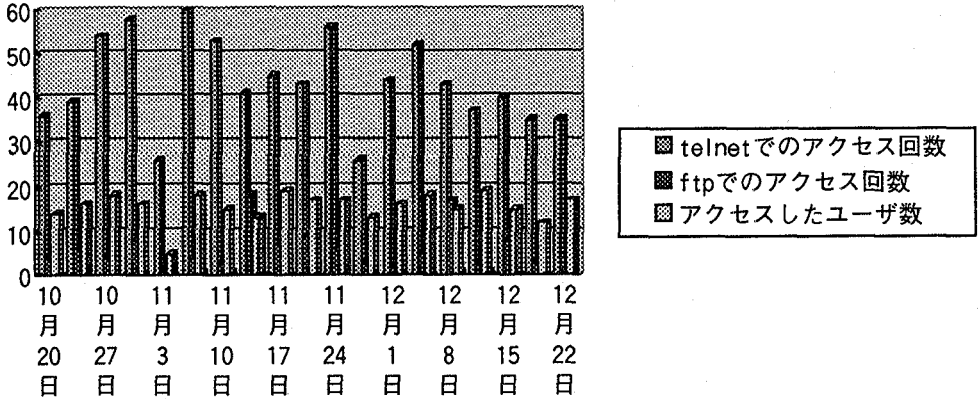
Oct/31 58 8 16
Nov/3 26 1 5
Nov/7 60 4 18
Nov/10 54 3 15
Nov/14 41 18 14
Nov/17 45 7 19
Nov/21 43 9 17
Nov/24 55 7 17
Nov/28 26 3 12
Dec/1 44 2 15
Dec/5 53 5 18
Dec/8 43 17 15
Dec/12 37 3 19
Dec/15 40 8 14
Dec/19 35 3 12
Dec/22 35 1 17
Dec/26 50 7 15
Dec/29 8 0 5
Jan/2 4 0 1
Jan/5 32 5 11

<<以下, 省略>>

(fourier) imai[6]

UNIX では毎日、最初にユーザがログインした時点から telnet/ftp の利用履歴がロギングされている。コマンド last は、任意のユーザによって last がタイプされる時点までのロギングされている当日の利用履歴を表示させる。ここではサンプリングしたい日の 23 時 59 分頃に last を cron から起動させることで利用状況の履歴を収集している。このようにして収集したデータに対して前述の perl スクリプトで処理した結果をグラフ化して図 IV-5 に示す。

表計算ソフトウェア (Excel 5.0) での集計では、telnet コマンドでのアクセスはサンプリングした期間では毎日平均 41.2 回であり、ftp コマンドでのアクセスは毎日平均 5.6 回であった。また、毎日の平均ユーザ数が 14.7 名であったので、fourier に対して一人あたり telnet で 2.8 回程度、ftp で 0.38 回程度アクセ



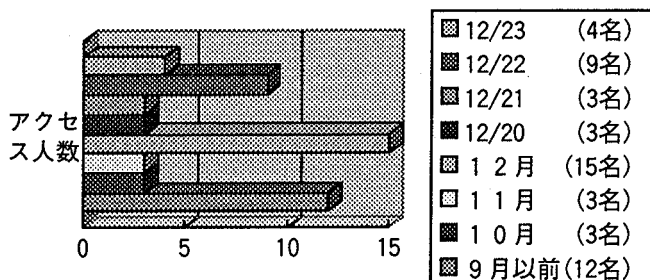
図IV-5 telnet/ftp ユーザのアクセス状況
(平成7年10月20日から12月22日まで)

スしていることになる。fourier に登録されていた学生の1日あたりのアクセスも平均して1.7回程度あったが、全体としては無視できる量であり、学生用メールサーバへの移行が順調に推移していることを示している。

WindowsPCではAL-MailやChameleon-Mailなどが利用でき、同様にMacintoshではEudoraと呼ばれるメールリーダーが利用できる。しかし、図IV-5のグラフからではWindowsPCやMacintoshなどのパソコンを利用してfourier上の電子メールサービスのみを利用しているユーザがどのような利用動向を示すかを判断することはできない。パソコンのメールリーダーでは、前述したpopperと交信することで、UNIX上の受信メールをパソコン上に簡単な操作で読み出すことが可能である。そこで、popperを利用してfourier上からメールを読み出しているユーザの動向を定量的に調べてパソコンユーザのメールサービス利用動向を可能な範囲で調査してみる。

パソコンユーザが電子メールを読み出す場合、popperが処理した時点でfourierの受信メールバッファ領域にサイズゼロの“.ユーザ名.pop"なるファイルが残ることになる。これらのファイルが生成された日時を集計してpopperの利用状況を調べることにした。結果を図IV-6に示す。メールの場合、誰か

らメールが送信されるか（先方が活発な利用者か、そうでないか）によって、個々のユーザの利用動向は様々である。また、メールを受信してくれる相手も必要である。受信メールへのアクセス記録は“ユーザ名.pop”なるファイルの生成日時によって把握できるが、送信メールをチェックするとなると、メール内容を調べて登録ユーザの誰が送信したメールかなどをチェックしなければならないことになるが、これは事実上不可能である。仮に可能であったとしても、明らかにプライバシーの侵害になるので、メール送信の動向を集計することは諦めざるをえない。システムが収集しているログ情報は個々のユーザが何時 fourier にアクセスしたかの履歴のみであり、それ以上のデータは常識的に考えても入手が困難であろう。



図IV-6 popper を利用したメール読み出しユーザ数

以上、収集し解析したデータを基に fourier の利用状況を簡単にまとめてみると、

1. fourier の登録ユーザであっても、あまり頻繁に利用していないユーザが最大3割程度存在している。
 2. fourier を電子メールサービス以外にも頻繁に利用しているユーザがやはり3割程度存在している。
 3. 電子メールサービスの場合、popper と呼ばれるデーモンソフトを利用しているパソコンユーザが3割から4割程度存在している。
- などとなっている。

IV-2 fourier のユーザ登録問題

fourier は SVR4 系の UNIX マシンであり、SVR4 の UNIX には “/usr/sbin/sysadm (以下, sysadm)” と呼ばれるシステム管理を会話的に行うソフトウェアが用意されている。sysadm の利点は多いが、ユーザ登録を会話的に行う場合ヒューマンエラーなどの混入を未然に防ぐ手立が見付けにくい。そこで、ユーザ登録にはバッチ処理型の操作で一括ユーザ登録を可能にする “/usr/sbin/useradd” や修正を行う “/usr/sbin/usermod” を活用する方法を選択している。バッチファイルの場合、どのような手順でユーザ登録を行ったかが、ドキュメントとして保存できる点も大きな特徴である [4]。

fourier 導入当時、経済学部は 3 学科構成であったため、各ユーザとなる経済学部教官を学科に分類して登録する、あるいは希望者のみの登録を行うなど、種々の検討を行った。最終的には『経済学部の伝統に馴染まない』などの指摘を受け、学科枠を設定しないユーザ登録としている。基本的に “/etc/passwd” のコメント部 (GCOS フィールド) にのみイニシャル氏名、学科の略称、大学名という形式で所属を明記することにした。例えば、今井であれば、

Y. Imai InfoSci Kagawa Univ.

なる個人情報が記述されている。ちなみに、現在は 4 つの学科に対して

経済学科	Economics
企業経営学科	BusinessAdmi
情報管理学科	InfoSci
地域社会システム学科	RegSocialSystem

という略称を用いている。個々のユーザの環境設定ファイル (各種ドットファイル) にはコマンドパスや最低限の alias, あるいはメール到着を知らせる設定などを用意していたが, popper を利用するメールユーザにとって UNIX マシンへ telnet などログインする可能性は相対的に低下する傾向にある。今後、ユーザ環境を再構築する際にはこのような利用状況を考慮すべきであろう。

パスワードの問題は UNIX において本質的であるが、いくつかの点で運用上の課題も提起している。fourier においても情報保護、システム保全などの security のため、パスワードを一定周期ごとに失効 (expiration) させ、パスワードの

再設定を要求する仕様となっている。パスワードの再設定自身はそれほど問題ではないと考えるが、UNIX に不慣れな多くのユーザにとって、パスワードの再設定を定期的に要求されることは、システムに対する信頼性を増すよりも返って利便性の低さを印象づける結果となっているようである。パスワードの失効期間を「設定せず」としていても自動的に失効期限が設定される事態に気づいた三谷はいくつかのテストを試みているが、パスワードを定期的に再設定するようユーザに要求する SVR4.2 の仕様は変更できない可能性が高いようである。積極的にこの機構を利用してパスワードの盗難対策に活用し security アップを行うか、あくまで便宜的に利用しパスワード失効時には、一度 “date0815” などの仮パスワードを設定し、再度、コマンド `passwd` を用いて元のパスワードに設定し直すか、などの対応は各ユーザに一任される。

他人にパスワードを知らせ、本人の代わりにメールの代読を依頼するなどの事態が発生していないのは経済学部の構成員の良識のなせる状況であろうと素直に喜んでいる。代読という行為自身を云々している訳ではない。個人のプライベートな情報を第三者が侵さないようにメールシステムを構成するためには、構成メンバー全員のコンセンサスをとる必要ある点を強調したいだけである。システム管理を行う上では、プライバシーの保護に最大限の留意を払っており、各ユーザの理解と協力がなければメールサービス自体を健全に運用できないと思われる。この点に関しても *fourier* の登録ユーザの教官各位に敬意を表するものである。

試験的に登録した学生ユーザの 1 名から、*fourier* の電子メールサービスを介して複数の教官に向けて特定の情報通知が唐突に行われたことは『晴天の霹靂』的な現象であった。学生は筆者達のゼミ生であり、先生方からは学生のこのような利用に対して何等かの対策を講じるよう指摘をいただいている。UNIX に関する知識があれば、“`/etc/passwd`” に登録されているエントリからユーザ名を特定でき、登録ユーザ全員にメールを送信することは容易である。このような学生の行為を予測できていなかったとは言え、速やかに学生専用のメールサーバを立ち上げ、試験的に登録していた学生ユーザも移行させる措置を講じなければならないと痛感した。電子メールは便利で研究費を圧迫しない情報伝達

手段であるが、半面モラルの徹底などもけっして疎かにできないという問題提起であった。管理者が試験的とは言え、経済学部メールサーバに学生をユーザ登録したことに対する責任が甚大である点を改めて感じている。

V お わ り に

事務系職員を試験的に fourier へユーザ登録し、メールサービスを経験していただいた成果が平成8年度には少しずつ現れそうな段階に到達してきた。事務系メールサーバ(kepler)も立ち上がり、電子メールを教官・事務官相互で積極的に交換し合う土壌が育ちつつある。一方、学生用の電子メールサーバ(現時点ではleibniz)も試験的な運用を開始し評価データを収集し始めている。経済学部メールサーバへの一時的なユーザ登録でいくつかの問題点を発生させながらも、電子メールサービスの利用者が確実に増加し、分散型電子メールサーバ体制の構築が目前に迫ってきた。

電子メールサービスがほぼ完成しつつある現在、fourier に対するいくつかの要望が寄せられ始めている。情報処理センターでのサービス方針もあって、CPUサーバやファイルサーバとしての能力をfourier自体に期待されていない反面、情報処理環境ではなくて情報通信環境の提供を目的とした学部固有の情報サーバとしての役割が期待されるようになった。例えば、

- 学部レベルのローカルニュースサーバ
- 学部レベルのWWWサーバ

などのサービスにfourierを利用できないかという意見や要望である。サービスを提供するソフトウェアの実装自体は比較的簡単であり、フリーウェアを活用すれば経費もゼロで実現可能である。fourierの運用方針や管理体制といった側面をどのように解決するかが先決課題となろう。

現在、1.1GBのディスクが6割程度の利用状況となっている。ユーザ数は恐らく今後大きくは増加しないだろうし、電子メールサービスだけを対象と考えれば基本的に現状維持でもよく、設備的にも問題はないと思われる。しかし、サービスを拡充すればそれに伴って、ディスクの増設と適切なバックアップ体制の確立とが今後の課題の1つとなってくる。過去に一度、fourierがディスク

クラッシュを起こした時はルーズなバックアップ体制が災いしたが、SEの中沢美弥子氏の適切な処置で事無きをえた。しかし、適切なバックアップ体制は不可欠であり、その意味でも fourier の管理体制をいかに確立するかも今後の課題と言えるだろう。

謝辞

fourier を含めたマシンサポートを御願っている SNES の中沢美弥子氏に感謝します。情報処理センターの瀬野芳孝氏、曾根計俊氏および丸山久美子氏には fourier の管理業務に関して日頃から何かと相談している。また、情報管理学科、中村邦彦教授、宍戸栄徳教授、富永浩之助教授および加藤大志朗講師にはいつも適切な助言をいただいている。記して謝意を表します。

参 考 文 献

- [1] 村井純・吉村伸監修 (1994) 『インターネット参加の手引き 1994 年度版』共立出版
- [2] 山口和紀監修 (1992) 『The UNIX Super Text(上)(下)』技術評論社
- [3] Wall, L. and Schwartz, R. (1990) "Programming Perl," O'Reilly and Associates (近藤嘉雪訳 (1993) 『Perl プログラミング』, ソフトバンク)
- [4] 今井慈郎・三谷宗子・本田道夫 (1995) "UNIX システム管理からみたプログラムとスクリプトとの機能比較", 『香川大学経済論叢』, 第 67 巻, 第 3・4 号