

# 九州大学 情報基盤センター

## 広報

学内共同利用版  
2006年 Vol. 6 No. 1

### 目次

#### 特集

九大ポータルMy. Kyushu-Uの試作

…………… 伊東栄典, 池田嗣徳, 亀岡謙一, 高木早智子 1

『e-ラーニング支援室』の設置について …………… 井上 仁 7

#### 解説

Mac OS X-コマンドライン・システム管理入門- …………… 内藤久資 11

#### 報告

平成17年度教育用システム統計 …………… 33

人事異動 …………… 34

編集後記 …………… 35

Computing and Communications Center  
**Kyushu University**

[www.cc.kyushu-u.ac.jp](http://www.cc.kyushu-u.ac.jp)

## 情報基盤センターサービス機器一覧

### ■教育用システム

---

ホスト計算機	ah.s.kyushu-u.ac.jp
WWWサーバ	www.cc.kyushu-u.ac.jp/ec/
ウェブメーラー(GraceMail)	https://secure.s.kyushu-u.ac.jp/

---

### ■KITE関連機器

---

教職員向けメールサーバ	mbox.nc.kyushu-u.ac.jp
-------------	------------------------

---

### ■リモートアクセスサービス

---

電話番号	制御手順	種別	通信速度	回線数
092-642-7341	PPP	モデム	最高56Kbps	64回線
		ISDN	同期64Kbps	92回線
		PHS (PIAFS)	32/64Kbps	32回線

---

# 九大ポータル My.Kyushu-U の試作

伊東栄典\* , 池田嗣穂 , 亀岡謙一 , 高木早智子

本稿では、当センターで試作している九大ポータル“My.Kyushu-U”について紹介します。なお、本稿を読まずとも、以下のページをアクセスして頂ければ利用できると思います。

My.Kyushu-U	<a href="http://my.kyushu-u.ac.jp/">http://my.kyushu-u.ac.jp/</a>
部局サイトの RSS 一覧	<a href="http://my.kyushu-u.ac.jp/rss/">http://my.kyushu-u.ac.jp/rss/</a>

本システムは試作段階ですので、改善の余地が沢山あります。利用者の皆様からのコメント、御意見を募集しております。御意見がある場合、下記の宛先にお送り下さい。

コメント送付先 [qportal@cypher.cc.kyushu-u.ac.jp](mailto:qportal@cypher.cc.kyushu-u.ac.jp)

## 1 はじめに

情報通信技術の発達と Web の広がりにより、Web ページは増加の一途をたどっています。九州大学内部でも多くの Web サイトが提供されており、多数の Web ページが存在しています。Google の検索ヒット数から判断すると、2006 年 2 月には 100 万以上のページが kyushu-u.ac.jp ドメイン内から提供されているようです。

九州大学のような総合大学では、学部・学府・施設・サークルなどの学内部局が存在し、部局のサイトも存在しています。九州大学の公式 Web サイトで提供されている、学部・大学院・研究所等の一覧を記載したページ (<http://www.kyushu-u.ac.jp/profile/faculty.html>) には、学内にある様々な部局やセンター等が提示されており、2006 年 2 月 17 日現在では 66 個の Web サイトへのリンクが存在しています。

多くの Web サイトやページがある場合、ページ閲覧者にとって二つの問題が発生します。一つは Web ページを巡回する手間であり、もう一つは情報過多による閲覧性の低下です。

九州大学の学生・教職員は所属部局の掲示情報を閲覧する他に、学内で共通するサイトの情報も閲覧する必要があります。例えば、九大トップページや図書館の情報は閲覧の必要性が高いでしょう。所属部局以外に、関連研究所や所属サークルといった、他サイトも閲覧しなければならない場合もあるでしょう。部局やサークルのお知らせは、そのサイトにしか掲載されない事が多いため、常に新鮮な情報を得るには複数のサイトを巡回しなければなりません。しかし、これは手間の掛かる作業です。

二つ目の問題は、情報過多による閲覧性の低下です。各サイトには多くのページが提供されているため、自分の必要な情報がどこにあるのか分かりにくく、また探し出すのも困難です。

---

\*情報基盤センター {itou,ikedata,kameoka,takagi}@cc.kyushu-u.ac.jp

また、上記の二つの問題は閲覧者に限ったものではなく、情報提供者側から考えても問題となります。提供者側は、知りたい方全員に「お知らせ」を伝えたいのですが、閲覧者を特定する手段がありません。かといって学内の全構成員にお知らせを連絡すると、部局は多数存在するわけですから、「お知らせ」がスパムメールのように増えて迷惑な情報過多を引き起こします。

以上の問題点の解決方法として、一元的に情報を提供するポータルサイトが考えられてきました。今回、学内の「お知らせ」を一元的に取りまとめて提供する九大ポータル“My.Kyushu-U”を試作しました。本稿では(試作版) My.Kyushu-U の機能について紹介します。

## 2 システムの概要

### 2.1 学内情報の一元化

ポータルサイトを構築するには二つの方法が考えられます。一つはデータベースを備えた統合情報提供システムを用意し、ここへ各部局のお知らせ情報を登録してもらう方法です。この方法では、統合情報提供システムの準備の費用が必要になります。また、各 Web サイトの情報提供者は統合情報提供システムへの情報登録の方法を新たに学習しなければならず、さらに自部局サイトへの情報更新に加えて、統合情報提供システムへの情報登録作業が発生してしまいます。

もう一つの方法は、分散して存在している Web 情報を、半自動的に抽出・統合して閲覧者へ掲示する方法です。この方法は、各 Web サイトの情報提供者は、何の手間も必要なく、既存の利用者にとっても閲覧方法の変更も不要です。ただし、情報統合システムの構築と保守には手間がかかります。

二つの方法について手間と費用を比較した結果、ポータルサイトの構築には後者の方法を採用しました [1]。

### 2.2 一元化する情報

次にどのような情報をポータルサイトへ一元化するべきかについて検討しました。各部局の Web サイトで提供されている情報は、二つの種類があると考えられます。一つは比較的長期間変更のない情報で、教員リスト、講義時間割、教室の位置、事務室への連絡方法などです。これらは年度中に変更されることはあまりありません。もう一つは、お知らせやニュースのような短期的な情報です。お知らせは随時提供されるため、頻繁に更新されます。

長期的な情報については、検索機能を用いて探して閲覧する想定しました。ただし、頻繁に閲覧される可能性があるページの場合は、そのページへのリンクをポータルサイト側に作成することにしました。リンクする場合は、定期的に URL 先のページの有無を調べる必要があります。

短期的な情報については即応性が重要であるため、随時ポータルサイト側にも提示する必要があります。そこで、お知らせの提示されている部位を切り出すプログラムを作成しました。これを定期的(一日に2回)に実行し、更新されていたらお知らせを切り出すようにしました。

切り出した「お知らせ」は、RSS 形式で保存しています。RSS とは RDF Site Summary の略で、XML でサイトの概要を記述するフォーマットであり、Netscape 社が 1993 年 3 月に公開したものです [2, 3]。最近では新聞サイトなどで、ニュース記事のヘッドラインを表現するの

に使われています。学内のお知らせもニュース記事の一つですから、RSS はぴったりの形式といえます。

## 2.3 お知らせ部分の抽出

各部署のサイト毎に、お知らせが存在するかどうかを手で調査しました。その結果、現在は43の部署でのお知らせが提供されていることが分かりました。次に、お知らせの提示されている部位を指定するXPathを、お知らせ提供サイト毎に一つ一つ手で調べました。手で調べたXPathを用いて、各部署サイトのお知らせの見出しを抽出します。抽出した情報を用いて、部署毎にRSS ファイルを自動生成するプログラムを作成しました。このプログラムを定期的に行うことで、新鮮なお知らせ情報を提示するためのRSS ファイル作成を行なっています。

## 3 My.Kyushu-U の利用方法

次にMy.Kyushu-U の利用方法を説明します。といっても利用はとっても簡単です。この説明を読まずとも、殆んど問題なく利用できるとおもいます。

### 3.1 起動画面

図1の左側に、Web ブラウザでMy.Kyushu-U (<http://my.kyushu-u.ac.jp/>) を表示させた場合の様子を示します。左側が起動時の画面です。図1の右側は、お知らせの一つをクリックした場合を表しています。中央に表示されているお知らせ項目の見出しをクリックすると、お知らせの詳細内容が表示する別のウィンドウが開きます。



(起動時)



(お知らせ表示)

図1: My.Kyushu-U

図1の左側の画面で分かるように、附属図書館が提供する「蔵書検索」と、大学評価情報室が提供している「研究者情報」への検索窓を置いています。また、頻繁に利用するサイトへのリンクも記載しています。

別の部局のお知らせを表示させるには、左側の部局名をクリックします。図2に、「情報基盤センター」のお知らせを表示させた場合を示しています。



図 2: 部局 (ここでは情報基盤センター) のお知らせ表示

### 3.2 表示部局のカスタマイズ

最初の起動時には、九大トップページと、計算機を接続したドメインのある部局のお知らせしか表示されません。他の部局のお知らせをいつも表示させたい場合には、「Custom」をクリックして、カスタマイズを行ないます。図3の左側に、カスタマイズ画面を示します。

カスタマイズ画面では、お知らせを表示可能な部局が一覧表示されます。表示させたい部局がある場合は、その部局にチェックを入れます。その後、[登録] ボタンをクリックすると、次からはチェックを入れた部局が My.Kyushu-U のトップに表示されるようになります。図3では、例として比較社会文化学府のお知らせを表示させてみました。



(表示部局の変更)



(カスタマイズ後)

図 3: 表示部局のカスタマイズ

### 3.3 RSS ファイルの利用

http://my.kyushu-u.ac.jp/rss/ を見ると、My.Kyushu-U が扱っている部局サイトの RSS が一覧表示されます。[RSS] と書かれたアイコンから、RSS ファイルへのリンク URL が取得できます。glucose 等の RSS リーダーを利用している場合は、こちらから必要とするサイトの RSS ファイルへの URL を取得下さい。



図 4: 各部局サイトの RSS 一覧

例として、RSS リーダーソフト “glucose2” ( http://glucose.jp/ ) を用いて、九大トップページのお知らせ RSS と、一つのお知らせ記事ページを表示させてみました。図 5 にその様子を示します。



図 5: 各部局サイトの RSS 一覧

## 4 おわりに

本稿では、情報基盤センターで試作した学内ポータルサイト “My.Kyushu-U” について紹介しました。まだまだ試作の域を出ておりませんが、学内の皆様が少しでも便利に使って頂けれ

ばと思います。

今後は、まず、Web サーバ間の連携を進めていきたいと思います。シラバスの教員名と教員 DB との連携、シラバス内の教科書と図書館の蔵書検索との連携など、様々な連携が可能でしょう。それらを進めていきたいと思います。

また、認証機構の導入も行ないます。学生の認証基盤は既に構築されているので、学生から利用者認証機構の導入を行ないます。また、当センターが学生向けに提供している Web Mailer “Grace Mail” (<https://secure.s.kyushu-u.ac.jp/mail/>) や、e-Learning システム “WebCT” (<http://webct.kyushu-u.ac.jp/>) など複数の Web 関連サービスシステムがあります。これらのシステムとの間で Single Sign On 機能も実現していきたいと考えています。

将来全学認証基盤が構築され、教職員の認証も可能になれば、研究者情報データベースの入力システムなども Single Sign On の連携範囲に入るでしょう。利用者認証機構が導入されれば、新着メールの件数表示なども可能になると思われます。今後は、これらの情報サービス機能を拡充していく予定です。

## 参考文献

- [1] 渦尾秀勝, 甲斐啓文, 伊東栄典, 廣川佐千男: ”学内ポータルサイト構築のための分散 Web 情報の一元化”, 情報処理学会 マルチメディア, 分散, 協調とモバイル (DICOMO 2005) シンポジウム論文集, pp.465-468, 2005.
- [2] RDF Site Summary(RSS)1.0, <http://web.resource.org/rss/1.0/spec>
- [3] RSS-サイト情報の要約と公開, <http://www.kanzaki.com/docs/sw/rss.html>
- [4] glucose2 : <http://glucose.jp/>



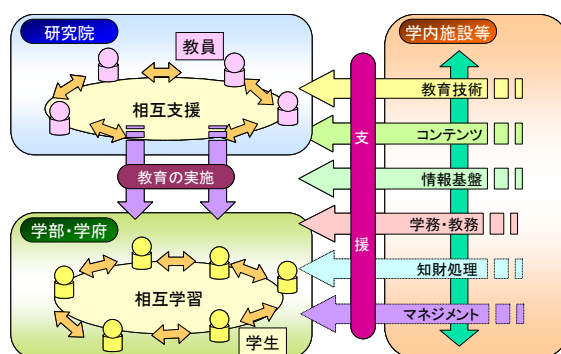
# 『eラーニング支援室』の設置について

井上 仁†

## 1 はじめに

情報基盤センターの学内向けのサービスの一つに、教育の情報化とその支援があります。従来からのサービスであった授業等で利用する PC やホスト計算機，メールシステム等の利用環境の提供に加えて，近年ではコース管理システム WebCT[1]や VOD システム[2]等のいわゆる eラーニングシステムを導入し，サービスを提供しています。

eラーニングを始めとする情報通信技術を利用した教育，いわゆる教育の情報化が組織的に実施され成功するためには，単に個々のサービスを提供するだけではなく，右図に示すように，教育の実施部局と学内の教育を支援する組織との連携と協力が必要です。さらには，教育を支援する組織間の連携と協力が必要です。しかしながら，教育の情報化のための学内支援体制は十分といえないのが現状です。



学内教育支援体制の概念図

そこで，eラーニングに関する学内支援を強化するために，情報基盤センター内に「eラーニング支援室」を設置しました。本支援室は「eラーニング」という名称をつけていますが，情報通信技術を利用した新しい教育全般を対象としています。

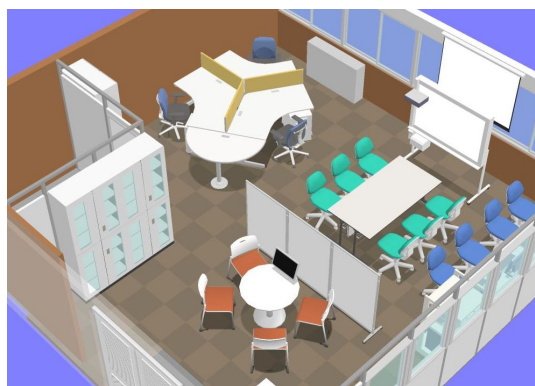
## 2 eラーニング支援室概要

今回設置したeラーニング支援室は，「支援業務」そのものと支援を行うための物理的な「施設」の二つの側面があります。「支援業務」は本支援室を設置する以前からすでに実施していた支援内容を今回拡大し明確にしました。「施設」は新たな支援に対応するための物理的な空間となります。

### 2.1 施設概要

eラーニング支援室は，情報基盤センターの2階にあり，基本的に平日 10 時から 17 時まで開室しています。

この施設では，コンテンツ作成のための機材の貸出し，コンテンツ作成のための場の提供，小規模な講習会・セミナーの開催，eラーニングに関す



eラーニング支援室 (イラスト提供：株式会社 内田洋行)

† 九州大学情報基盤センター jin@cc.kyushu-u.ac.jp

る書籍や資料の公開，コンサルティング，教職員のコミュニティの場の提供を行っています。

## 2.2 支援業務概要

「支援」といった場合，一般には，(1)高度な技術が必要な専門的な業務を遂行する，(2)業務自体は単純であるけれど生産性を高めるために一括集中して業務を代行する，(3)各組織が専門的な技術を身につけ成長するための支援を行う，等の形態が考えられます。

e ラーニング支援室で対象にしている支援の多くは(3)に該当します。e ラーニングに限らず新たな技術を導入する場合は(1)のような支援体制が必要ですが，情報通信技術を利用した新たな教育においては，教育実施の主体となる部局の教職員がある程度の専門的な技術を身につける必要があると考えています。もちろん将来的には，(2)を実施する支援組織が学内に必要になります。

すなわち，e ラーニング支援室では以下の業務を行います。

- ・ コンテンツ作成のための機器の貸出しと場所の提供
- ・ 教育研究情報公開のための事業支援
- ・ e ラーニング技術の普及ならびに活用支援
- ・ 国内外の e ラーニング動向調査と情報提供

### ■ コンテンツ作成のための機器の貸出しと場所の提供

講習会や授業を録画し，ストリーミング配信あるいは VOD で配信するためには，録画のための機材，編集機器，配信サーバが必要になります。

e ラーニング支援室では，ビデオカメラ，ビデオスイッチャー，RGB スwitchャー，音声ミキサ，CAC (Contents Auto Creator)[2]，各種ケーブル等の機材の貸出しと利用に関するサポートを行っています。また，HelixServer や MediaBase による配信も行っています。



### ■ 教育研究情報公開のための事業支援

自由で制限のない「知」へのアクセスを促進することを目的にしている OCW(Open Course Ware)プロジェクトが，世界規模で進行しています。九州大学でも，本学で開講されている授業のシラバス・カレンダー・授業資料等を公開する九大 OCW 事業を開始しました[3]。e ラーニング支援室では，九大 OCW の企画・運営・Web サイトの管理を支援しています。

また，VOD サーバによる教育研究情報発信事業も支援しています。

### ■ e ラーニング技術の普及ならびに活用支援

教育の情報化が組織的に実施され成功するためには，教育の実施部局と学内の教育を支援する組織との連携と協力が必要です。教育支援としては，情報通信基盤技術，コンテンツ作成，情報通信技術を活用した教育手法，学務・教務システム，知財処理，マネジメント等があります。e ラーニング支援室では，主に情報通信基盤技術の提供と技術指導を対象として

います。なお、他に関しては非専門ではありますが、蓄積した経験に基づく簡単な助言は可能と考えております。

具体的には、部局における e ラーニング実施のためのコンサルティング、コンテンツ作成のための技術支援、講習会の開催、部局のファカルティ・ディベロップメントの企画や講演等を行っています。

### ■ 国内外の e ラーニング動向調査と情報提供

国内外の大学等で実施されている e ラーニングの動向調査と情報提供のために、各機関が発行している資料を収集し閲覧できるようにしています。また、e ラーニングに関する研究会・学会・研修会・展示会等が開催されていますが、日程の都合や遠方で開催されるために参加できないことがあります。そこで、e ラーニング支援室で収集した資料を閲覧できるようにしています。



また、学内で実践されている e ラーニングの事例を学内外に紹介していきます。

## 3 支援の例

ここでは、情報基盤センターが過去に実施してきた支援業務で e ラーニング支援室の対象となるものや現在支援を行っているものをいくつか紹介します。

### VOD 配信支援

研究戦略企画室の企画による 21 世紀 COE プログラム拠点形成発表会の VOD 配信に対して、機器の提供と技術支援、発表会当日の技術補助、VOD サーバによるコンテンツ配信を行いました。なお、撮影とコンテンツ作成のための人件費等は、主催者側で負担していただきました。

研究戦略企画室の企画による日本学術振興会特別研究員説明会の VOD 配信に対して、機器の提供と技術支援、事前準備、VOD サーバによるコンテンツ配信を行いました。撮影とコンテンツ作成のための人件費は、主催者側で負担していただきました。

総合理工学研究院の主催による総理工セミナーの VOD 配信に対して、機器の提供と VOD サーバによるコンテンツ配信を行いました。本支援では、セミナー開催時には機器の貸出しのみを行い、総合理工学研究院のスタッフにより撮影・編集をしていただきました。

### ファカルティ・ディベロップメント(FD)での講演

総合理工学府の FD において、九州大学における e ラーニングの実践事例について講演しました。また、システム情報科学府の FD においては、九大 OCW に関して説明しました。ある学部の次回開催の FD で e ラーニングを話題にする予定であり、現在準備をすすめています。

### 「現代的教育ニーズ取組支援プログラム」への参加

平成 16 年度の文部科学省現代的教育ニーズ取組支援プログラムに採択された「WBT による医療系統合教育」プロジェクトに参加し、e ラーニング実施を支援しています。

### 「魅力ある大学院教育」イニシアティブの支援

平成 17 年度の文部科学省「魅力ある大学院教育」イニシアティブに採択された総合理工学府の「ものづくり型実践的研究人材の戦略的育成」プロジェクト実施において、e ラーニング実施のための支援を行っています。

### WebCT 導入支援

各部局における WebCT の組織的な導入にあたって、導入のためのコンサルティング、講習会の開催、コンテンツ作成の助言等を行っています。個人的に WebCT を利用される教員に対しても、単なる操作方法の説明だけではなく、実践事例の紹介や授業において効果的な利用法の提案等を行っています。

### システム導入支援

学内の各種教育支援のためのシステムの導入にあたり、技術的な助言等を行っています。

## 4 おわりに

本稿では、情報基盤センターに新たに設置した e ラーニング支援室について簡単に紹介しました。e ラーニング支援室は業務を開始したばかりですが、学内の教育の情報化に少しでも役に立てればと考えています。Web サイト(<http://el.cc.kyushu-u.ac.jp/>)でも、さまざまな情報を提供していますので、ぜひご覧ください。また、情報基盤センターにお越しの際には、ぜひ 2 階の e ラーニング支援室にもお立ち寄りください。

## 参考文献

- [1] 特集「e ラーニングシステム WebCT ー導入と利用の展開ー」, 井上仁, 九州大学情報基盤センター広報学内共同利用版, Vol.3, No.2, 2003.
- [2] 「Contents Auto Creator による VOD コンテンツ作成と MediaBase による配信」, 井上仁, 多川孝央, 九州大学情報基盤センター広報学内共同利用版, Vol.5, No.3, 2006.
- [3] 九州大学 OCW <http://ocw.kyushu-u.ac.jp/>

# Mac OS X

– コマンドライン・システム管理入門 – \*

内藤 久資

名古屋大学多元数理科学研究科

naito@math.nagoya-u.ac.jp

7回にわたって書き続けてきた Mac OS X の解説の締めくくりとして、今回は Mac OS X の「コマンドライン」ツールの解説をしたい<sup>1</sup>

Apple ユーザの多くは、その GUI の操作感のよさや、Mac OS X の安定性・安全性などを理由に Mac OS X を利用しているのだと考える。Mac OS X がシステムとして安定している理由の多くは、Mac OS X の基本システムが BSD と呼ばれる UNIX であることに依存している。Mac OS X には BSD システムとしてのコマンドラインでのツールが数多く含まれているため、それらを利用することにより、UNIX の手法を使った利用形態が可能である。コマンドラインでの利用法は、特にシステム管理に有効なものが多い。BSD のツール群そのものの解説は FreeBSD などの解説書などにゆずり、Mac OS X 特有のツール群のコマンドラインでの利用法に焦点を絞って解説する。

## 17 コマンドラインの利用法と BSD ツール群

はじめに、コマンドラインで Mac OS X を利用するための方法を簡単に述べ、Mac OS X にインストールされている BSD ツール群を簡単に紹介しておこう。

### 17.1 コマンドラインの利用法

「コマンドラインで Mac OS X を利用する」とは、対話型シェルを利用して、コマンドを入力して Mac OS X を利用するということである。そのため、最初に対話型シェルが利用できる「仮想ターミナル」を起動する必要がある。

Mac OS X で標準的に利用できる仮想ターミナルソフトウェアとしては以下のものがある。

- 「ターミナル」アプリケーション。  
「アプリケーション」フォルダの中の「ユーティリティ」フォルダにある。「ターミナル」アプリケーションは、日本語表示、UTF-8 文字表示にも対応している。
- 「X11」の仮想ターミナルである xterm, kterm, uxterm など。  
「X11」は、Tiger Install Disc を利用するとインストールすることができる。xterm は日本語表示には対応していないが、uxterm は UTF-8 に対応している。古典的な日本語仮想ターミナル

\*この記事は名古屋大学情報連携基盤センターニュース (Vol. 5, No. 1, pp. 16–38) に掲載されたものを加筆・訂正したものである。「【注】」とある部分は、九州大学広報に掲載するにあたって加筆した部分である。

<sup>1</sup>この原稿を執筆している 2005 年 11 月現在、Mac OS X の最新リリースは 10.4.3 である。この解説は、特に断らない限り、Mac OS X 10.4.3 に沿うものをご理解いただきたい。

【注】 2006 年 3 月現在の最新リリースは 10.4.5 である。



である kterm は Apple から配付されている X11 のパッケージには入っていないので、個別にインストールする必要がある。

また、「システム環境設定」の「共有設定」で「リモートログイン」を有効にしている場合には、ssh を利用してネットワーク経由でこれらの対話型シェルを利用することも可能である。

さて、仮想ターミナルを開くとすぐに対話型シェルが起動する<sup>2</sup>。仮想ターミナルを利用するには次のことに注意が必要である。

- Aqua 環境では「デスクトップ」などと日本語で表示されていたフォルダ名は、シェル環境では“Desktop” などとなり、Aqua の英語環境で表示されるフォルダ名がディレクトリ名となっている。
- ファインダ環境で「フォルダ」として表示されるものは、UNIX システムでのディレクトリであるが、「エイリアス」は UNIX システムでのシンボリックリンクではない。
- ファイル名に含まれる日本語が表示できない。

Mac OS X のファイルシステムでは、ファイル名には UTF-8 エンコーディングが利用されている。「ターミナル」アプリケーションを使って `ls` コマンドで日本語を含むファイル名を正しく表示させるには `-v` オプションをつければよい<sup>3</sup>。

なお、仮想ターミナルを起動した時点では、ログインしているユーザの権限でコマンドが動作することになるが、「管理者権限」が必要な場合には `sudo` コマンドを利用して、管理者モードに移行する必要がある。

#### 【sudo の利用法】

- `sudo COMMAND`  
COMMAND を管理者権限で実行する。実行後は一般ユーザ権限に戻る。
- `sudo -s`  
管理者モードに移行する。一般ユーザモードに戻るには `exit` コマンドを実行する。

なお、`sudo` を利用して管理者モードに移行できるユーザは「管理者」として登録されたユーザに限られる。この設定を変更するには `sudo` コマンドの動作を規定している `sudoer` ファイルを `visudo` コマンドを利用して書き換えればよい。

## 17.2 BSD ツール群

冒頭にも述べた通り、Mac OS X は、その裏に BSD システムが存在しているので、基本的な BSD システムのツール群の多くは、デフォルトでインストールされている。どのようなツール群（アプリケーション）がインストールされているかを知るためには、`/usr/bin`、`/usr/sbin`、`/usr/X11R6/bin` などのディレクトリを覗けばよい。そこには、BSD ユーザならば見たことがあるツール群が一通りそろっている。

しかしながら、ごく一部に通常の BSD システムと同じ名前であるが、使用時に注意すべきコマンドがいくつかあるので、それだけはメモしておこう<sup>4</sup>。

<sup>2</sup>Tiger のデフォルト設定では `bash` が起動するが、ユーザのログインシェルの設定を変更すると `tcsh` などの他のシェルを利用することも可能である。

<sup>3</sup>コマンドの引数に日本語を含むファイル名を指定することも可能である。

<sup>4</sup>これは筆者がハマった例であり、他にも異なった挙動をするコマンドが存在する可能性がある。

- /usr/bin/tar コマンド.  
/usr/bin/tar の他に /usr/bin/gnutar がある. /usr/bin/tar を使うと, 既に存在しているディレクトリの上に上書きができない. 通常の tar コマンドと同じ動きをするのは /usr/bin/gnutar である.
- /bin/cp, /bin/rm, /bin/mv などのファイル操作コマンド.  
これらのコマンドを Mac OS X のアプリケーション, またはそれらが作成したファイルに利用するのは危険である<sup>5</sup>. なぜなら, Mac OS X のアプリケーションやそれらが作成したファイルには「リソース」と呼ばれる属性データが付随している場合があり, これらのファイル操作コマンド群ではリソースを含んだファイル操作ができない. リソースを含んだファイルに対するファイル操作を行うには「開発環境」に付属するコマンド群(後述)を利用する必要がある.

### 17.3 各種のサービス

Mac OS X のネットワークサービスなどに利用されているソフトウェアの多くも, BSD などで標準的に利用されているものが流用されている. それらのコマンドラインなどでの管理方法はそれぞれのウェブページなどを参照すればよい.

#### プリンティングシステム

プリンティングシステムは CUPS が利用されている (cf. [8]).<sup>6</sup>

#### ネットワークサービス

ネットワークサービスの制御には xinetd が利用されている (cf. [9]).

#### ウェブサーバ

ウェブサーバは Apache が, サブレットコンテナには Apache Tomcat が利用されている (cf. [10]).

## 18 Mac OS X 特有のコマンド群

以下では Mac OS X に特有なコマンドで, システム管理等に有効なものを順に紹介しよう. それらは GUI ベースのアプリケーションと同等の動きをするものが多く, これらを組合せることにより, システム管理の自動化を実現することができる.

なお, ここで紹介するコマンドの使用例は, 各コマンドの機能の一部であり, 詳細は man コマンドを利用してオンラインマニュアルを参照して頂きたい.

### 18.1 ディスクユーティリティ

「ディスクユーティリティ」では以下の操作が可能である.

- ディスクのフォーマット, パーティションの作成, (ソフトウェア)RAID の設定, ディスクの検証などのディスクの管理.
- ディスクイメージの作成と復元.

<sup>5</sup>通常のファイルであれば問題はない.

<sup>6</sup>【注】BSD システムで広く利用されている lpd とは異なる, 新しいプリンティングシステムである.

これらの操作に対応するコマンドは `hdiutil` と `diskutil` である。 `diskutil` はローカルディスクの操作を行うコマンドであり、 `hdiutil` はディスクイメージの操作を行うコマンドである。

### 18.1.1 ディスクの操作

`diskutil` を用いるとローカルディスクに対して次の操作を行うことができる。

- ディスク情報の取得。
- ディスクのマウントとアンマウント。
- 取りだし可能なディスクの取りだし。
- ディスク名（ボリューム名）の変更。
- HFS+ journing の有効化・無効化。
- ディスクのチェックとリペア。
- ディスクパーミッションのチェックとリペア。
- ディスクやボリュームの消去、再フォーマット。
- ディスクパーティションの作成。
- RAID の作成・チェック。

#### 【使用例】

ディスクのリストの取得

`diskutil list` を実行すると以下のような結果を得る。

```
/dev/disk0
#:                type name      size      identifier
0: Apple_partition_scheme *232.9 GB disk0
1:  Apple_partition_map  31.5 KB   disk0s1
2:  Apple_HFS Second  232.8 GB  disk0s3
/dev/disk1
#:                type name      size      identifier
0: Apple_partition_scheme *149.1 GB disk1
1:  Apple_partition_map  31.5 KB   disk1s1
2:  Apple_Driver43        28.0 KB   disk1s2
3:  Apple_Driver43        28.0 KB   disk1s3
4:  Apple_Driver_ATA      28.0 KB   disk1s4
5:  Apple_Driver_ATA      28.0 KB   disk1s5
6:  Apple_FWDriver        256.0 KB  disk1s6
7:  Apple_Driver_IOKit    256.0 KB  disk1s7
8:  Apple_Patches         256.0 KB  disk1s8
9:  Apple_HFS MacOSX   149.0 GB  disk1s9
/dev/disk2
#:                type name      size      identifier
0: Apple_partition_scheme *152.7 GB disk2
1:  Apple_FWDriver        128.0 KB  disk2s1
2:  Apple_partition_map    8.0 KB    disk2s2
3:  Apple_HFS Data         152.7 GB  disk2s3
```

ここに表示されたのは、 `diskutil` が扱うことができるローカルに接続されたディスクのリストと、そのディスクのパーティション情報である。 `diskutil` の各動作で `device` と指定されているところには、 `/dev/disk0` などの各ディスクを指定する「デバイス名」を指定する。ここには「ディスクユーティリティ」でディスクとして表示されるものが全てリストされていることがわかる。



## ディスク情報の取得

`diskutil info /dev/disk1` を実行する。すなわち「デバイス」`/dev/disk1` の情報を取得すると以下のような結果を得る。

```
Device Node:          /dev/disk1
Device Identifier:    disk1
Mount Point:
Volume Name:

Partition Type:      Apple_partition_scheme
Bootable:             Not bootable
Media Type:           Generic
Protocol:             ATA
SMART Status:        Verified

Total Size:          149.1 GB
Free Space:           0.0 B

Read Only:            No
Ejectable:            No
OS 9 Drivers:         Yes
Low Level Format:     Not Supported
Device Location:     "A (upper)"
```

ここに表示されたのは、`/dev/disk1` のディスク情報であり、総容量、空き容量などが表示されている。

また、`diskutil info /dev/disk0s3` を実行すると、`/dev/disk0` の第3パーティションの情報を得ることができる。

```
Device Node:          /dev/disk0s3
Device Identifier:    disk0s3
Mount Point:          /Volumes/Second
Volume Name:          Second

File System:          Journaled HFS+
                      Journal size 24576 k at offset 0x749000
Owners:                Enabled
Partition Type:       Apple_HFS
Bootable:              Is bootable
Media Type:            Generic
Protocol:              ATA
SMART Status:         Verified
UUID:                 00ECF16A-5DA2-3D93-A342-BCC8EF04B384

Total Size:           232.8 GB
Free Space:            158.9 GB

Read Only:             No
Ejectable:             No
Device Location:      "B (lower)"
```

ここで利用した「デバイス名」だが、マウントされているディスクのデバイス名を取得するには `df` コマンドを利用すればよい。実際 `df -lkh` を実行すると

Filesystem	Size	Used	Avail	Capacity	Mounted on
<code>/dev/disk1s9</code>	149G	112G	37G	75%	<code>/</code>
<code>/dev/disk0s3</code>	233G	69G	164G	30%	<code>/Volumes/Second</code>
<code>/dev/disk2s3</code>	153G	137G	16G	90%	<code>/Volumes/Data</code>

という結果を得る。この結果から、起動ディスク（マウントポイントが `/` のデバイス）のデバイス名は `/dev/disk1s9` であることがわかる。

## ディスクの検証

`diskutil verifyVolume /dev/disk0s3` を実行すると、`/dev/disk0s3` のディスクの検証ができる。

```
Started verify/repair on volume disk0s3 Second
Checking HFS Plus volume.
Checking Extents Overflow file.
Checking Catalog file.
Checking Catalog hierarchy.
Checking Extended Attributes file.
Checking volume bitmap.
Checking volume information.
The volume Second appears to be OK.
Mounting Disk
Verify/repair finished on volume disk0s3 Second
```

このように「ディスクユーティリティ」を用いたディスクの操作は、全て `diskutil` を利用して行うことができる。`diskutil` の機能のうち最も有用なものは「ディスクのクリア」である。これは、ディスクの再フォーマット後にディスク全体にランダム（または全て0）のデータを書き込む操作であるが、数百 GB のディスク全体のデータ書き込みは長時間が必要となる。このとき「ディスクユーティリティ」を利用してしまうとログアウトができなくなるが、この `diskutil` を利用することにより、「バックグラウンドでジョブを実行」<sup>7</sup>すれば、ログアウトを行ってもコマンド実行は継続されているので、長時間にわたるジョブの実行が可能になる。

なお、XServe などのオプションとして販売されている「ハードウェア RAID カード」の設定は、この `diskutil` ではなく `megaraid` コマンドで行う必要がある。

### 18.1.2 ディスクイメージの操作

`hdiutil` を用いると、以下のようなディスクイメージの操作を行うことができる。

- ディスクイメージの作成。
- ディスクイメージのマウント・アンマウント・検証。
- ディスクイメージのフォーマット変換。
- ディスクイメージの光学ディスクへの書き込み。

#### 【使用例】

ディスクまたはフォルダのディスクイメージの作成

```
hdiutil create -srcfolder directory target.dmg
または
hdiutil create -srcdevice /dev/disk0s3 target.dmg
```

を実行する。前者はフォルダ（フォルダ名 `directory`）からディスクイメージを作成する例であり、後者はデバイス名 `/dev/disk0s3` を持つディスクからディスクイメージを作成する例である。ともに `target.dmg` は作成するディスクイメージの名称である。

なお、作成されるディスクイメージのフォーマットは、デフォルトは UDZO (UDIF zlib-compressed) であり、読み取り専用のディスクイメージが作成される。

<sup>7</sup>コマンドの末尾に `&` をつけて実行する。

## 空のディスクイメージの作成

```
hdiutil create -size 1m target.dmg
```

または

```
hdiutil create -size 1m -type SPARSE target.dmg
```

を実行する。いずれの場合にも `-size` オプションでディスクイメージのサイズを指定している。前者はディスクイメージサイズが 1M バイトのものを作成しているが、後者は 1M バイトまで拡張可能なものを作成している。

なお、`create` オペレーションでは以下のオプションも利用可能である。

- `-fs` フォーマットを指定する。HFS+などを指定可能。
- `-volname` ボリューム名を指定する。
- `-encryption` 暗号化されたディスクイメージを作成する。デフォルトの暗号化方式は 128 ビット AES (Advanced Encryption Standard) である。

## ディスクイメージのマウント

```
hdiutil attach target.dmg
```

を実行する。マウントされたときの名称はディスクイメージのボリューム名となる。

## ディスクイメージのアンマウント

```
hdiutil detach /dev/disk1s3
```

を実行する。アンマウントの対象となるディスクイメージはボリューム名ではなく、デバイス名を指定する必要がある。デバイス名を調べるには `df` コマンドを利用する。

## ディスクイメージのフォーマット変換

このためには

```
hdiutil convert -format UDRW -o new.dmg old.dmg
```

を実行する。old.dmg のフォーマットを変換し、new.dmg として保存する。この例で指定している UDRW とは、無圧縮の読み書き可能なフォーマットである。

## ディスクイメージから CD-R などを作成

```
hdiutil burn source.dmg
```

を実行する。`-noeject` オプションを指定した場合には、CD-R 作成後にマウントが実行される。

【より高度な使用例】 `hdiutil` を利用すると、「ホームフォルダ」の内容を定期的にディスクイメージとしてバックアップすることが可能となる。例えばホームフォルダが `/Users/naito` にあり、そのデータを「2 台目のディスク」`Second` にディスクイメージとして保存するためには、以下のようなシェルスクリプトを作成し、それを実行すればよい。

```
#!/bin/sh -f
TARGET="/Volumes/Second"
/bin/mv -f ${TARGET}/backup.dmg ${TARGET}/backup_old.dmg
/usr/bin/hdiutil create -srcfolder /Users/naito ${TARGET}/backup.dmg
```

さらに、このコマンドを cron を利用して、深夜に定期的に行うように設定すれば、無人でバックアップの作成が可能になる。

なお、cron を設定するには crontab コマンドを利用して、crontab と呼ばれる定時実行のためのデータファイルを作成すればよい。

### 18.1.3 データのコピー

ディスクイメージなどにバックアップしたデータをフォルダや新規ディスクにコピーするには ditto を利用する。ditto ではリソースも同時にコピーされる。

#### 【使用例】

フォルダ間のコピー

```
ditto srcfolder dstfolder
```

とすれば、srcfolder の中身を dstfolder にコピーできる。これを応用すればディスクイメージからのデータ復元が可能になる。

フォルダなどのデータのアーカイブ

```
ditto -c srcfolder dstfile
```

とすれば、srcfolder の中身を CPIO フォーマットで単一のファイル dstfile にコピーできる。

アーカイブからのデータの抽出

```
ditto -x srcfile dstfolder
```

とすれば、CPIO フォーマットのアーカイブファイルから dstfolder へデータの展開ができる。

なお、pax コマンドを利用することで CPIO フォーマットのアーカイブを作成することも可能であるが、pax コマンドはリソースまでは保存できない。

リソースまで含んでアーカイブを作成したい場合には、フリーウェアの hfspax を利用する方法がある<sup>8</sup>。

#### 【使用例】

データのアーカイブ

```
hfspax -w -f foo.pax .
```

カレントディレクトリを foo.pax にアーカイブする。用いられるアーカイブのフォーマットは、デフォルトでは ustar 形式である。<sup>9</sup>

アーカイブの展開

---

<sup>8</sup>hfspax はデフォルトではインストールされていない。

<sup>9</sup>-x オプションを用いれば cpio 形式も利用できる。

```
hfspace -r -v -f foo.pax
```

foo.pax をカレントディレクトリに展開する。  
アーカイブの中身の表示

```
hfspace -v -f foo.pax
```

foo.pax の内容を表示する。

UNIX のアーカイブ・ダンプコマンドを利用できない理由はリソースフォークがファイルとして認識できない点にあった。そのため、リソースフォークをファイルとして分離することが可能であれば、その後に UNIX のアーカイブコマンドでアーカイブすることが可能である。そのような方法を用いたのが以下の手順である。

ここでは、/tmp/FOO ディレクトリに(リソースフォークを持つファイルの代表例として)Microsoft Word の実行形式をコピーして、それを tar でアーカイブし、さらに、別のディレクトリで展開してみよう。

リソースを含めてコピーする

```
CpMac /Applications/MyApps/Microsoft Office X/Microsoft Word  
/tmp/FOO
```

これにより、Microsoft Word というファイルが /tmp/FOO に(リソースフォークつきで)コピーされた。この時 ls コマンドで見ることができるファイルの情報は「データフォーク」に関する情報であり、「リソースフォーク」はファイルシステムのカタログ B ツリーに格納されているので、UNIX レベルのコマンドでそれらを見ることはできない。

リソースを分離する

```
SplitForks /tmp/FOO
```

SplitForks コマンドはリソースフォークをファイルとして抽出するコマンドであり、ディレクトリ名を引数として指定すると、そのディレクトリ以下のすべてのファイルを対象にリソースフォークの抽出を行う。その実行結果として、\_Microsoft Word という新しいファイルが作成される。これが Microsoft Word に対応するリソースフォークであり、リソースフォークをファイルとして抽出できたことになる。

リソースごとアーカイブする

```
tar cvf FOO.tar ./FOO
```

すでにリソースフォークが FOO 以下にファイルとして存在するので、tar コマンドでリソースフォークまでアーカイブできる。

以上で、/tmp/FOO 以下をリソースフォークを込めてアーカイブできた。

逆に、分離されたリソースフォークを「結合する」ためのコマンドは /System/Library/CoreService/FixupResourceForks である。したがって、上で作成した tar アーカイブを展開して FixupResourceForks コマンドでリソースフォークを結合すれば、リソースフォークを込めてファイルアーカイブを完全に展開したことになる。その方法は次の通りである。

リソースを結合する

```
/System/Library/CoreServices/FixupResourceForks /tmp/FOO
```

この FixupResourceForks は指定されたディレクトリ以下のリソースフォークを結合する。

また、上のようにして CpMac でコピーしたファイルを完全に消去するためには、一旦 SplitForks でリソースフォークを分離して、データフォークとリソースフォークの両方を消去する必要がある。<sup>10</sup> なお、分離されたリソースフォークのままでも Mac OS X の動作には何ら支障はない。

## 18.2 ユーザ情報の操作

新規ユーザの作成、ユーザの消去などのユーザ情報の操作には NetInfo 関連のコマンド群を利用する。NetInfo の基本的な解説や、LDAP などその他のネーミングサービスに関しては、以前の解説 [4, Section 12] を参照して頂きたい。ここでは、NetInfo を利用した新規ユーザの作成などのユーザ設定を解説する。

### 18.2.1 既存のユーザ情報を閲覧する

通常の UNIX システムであれば、ユーザ情報は /etc/passwd ファイルや NIS, LDAP などのネーミングサービスから取り出すことができる。Mac OS X ではローカルなユーザ情報は NetInfo データベースに格納されている。初めに既存のユーザ情報を NetInfo データベースから取り出してみよう。

#### 【ユーザ情報を抽出する】

**nidump** コマンドを利用する **nidump** コマンドは NetInfo データベースの中身を（古典的な UNIX password の形式で）取り出すためのコマンドである。ユーザ情報を **nidump** を利用して取り出すには、取り出すべきデータベースの名称（ユーザデータベースは **passwd**）と、対象のドメイン（ローカルであれば“.”）を指定して **nidump** コマンドを実行すればよい。

```
nidump passwd .
```

その結果は

```
nobody:*:-2:-2::0:0:Unprivileged User:/dev/null:/dev/null
root:*:0:0::0:0:System Administrator:/var/root:/bin/tcsh
naito:*****:501:20::0:0:Hisashi NAITO:/Users/naito:/bin/tcsh
```

となる。（一部省略してある）

**niutil** コマンドを利用する より汎用的な NetInfo データベースの操作コマンドである **niutil** を利用することもできる。そのためには以下の手順をとる。

**NetInfo** データベース全体の項目リストを取得する **niutil** にはドメイン名と「パス名」を指定する必要がある。パス名とは NetInfo データベース上の「位置」のことであり、データベース全体は「ルート」である“/”で指定される。

```
niutil -list . /
```

<sup>10</sup> どうして RmMac というコマンドが存在しないのが不思議である。

その結果は

```
1      users
2      groups
3      machines
4      networks
5      protocols
6      rpcs
7      services
8      aliases
9      mounts
10     printers
66     locations
```

となる。

NetInfo の **users** データベースの項目リストを取得する 次に「ユーザ情報」が格納されている部分 **users** の項目リストを表示してみよう。今度はパス名として **/users** を指定する。

```
niutil -list . /users
```

その結果は

```
11     nobody
12     root
52     naito
```

となる。(一部省略)

個別のユーザの情報を取得する 個別のユーザの情報を取得するには、そのユーザのデータを対象に、**-read** を実行すればよい。

```
niutil -read . /users/naito
```

その結果は

```
picture: /Library/User Pictures/Animals/Butterfly.tif
_shadow_passwd:
hint:
uid: 501
_writers_passwd: naito
realname: Hisashi NAITO
_writers_hint: naito
gid: 20
home: /Users/naito
name: naito
_writers_tim_password: naito
_writers_picture: naito
shell: /bin/tcsh
sharedDir: Public
generateduid: BDF7CE12-0913-11D8-AD5E-000A9599B522
authentication_authority: ;ShadowHash;
passwd: *****
```

上記からわかるように、このデータベースを書き換えることができれば、ユーザ情報を変更したり、新規にユーザを作成、既存のユーザの削除が可能である。

## 18.2.2 ユーザの作成・削除など

### 【新規ユーザの作成】

niload コマンドを利用する niload コマンドは nidump コマンドの「逆操作」を行うものである。新規に“testuser”を作成するには、UNIX flat ファイル形式のユーザ情報を作成しておく。つまり、以下の1行が書かれたファイルを作成する。<sup>11</sup>

```
testuser:*****:502:20::0:0:TestUser:/Users/testuser:/bin/tcsh
```

このファイルを（例えば）newuser.txt として niload コマンドを実行すればよい。

```
sudo niload -v -m passwd . < newuser.txt
```

ここで、新規にユーザを追加するためには管理者権限が必要なため、sudo を経由して実行していることに注意しよう。なお、-m オプションは、既存のデータベースに追加することをあらかず。その結果は

```
1 items read from input
Netinfo /users contains 24 items
Processing input item:
_writers_passwd: testuser
change: 0
class:
expire: 0
gid: 20
home: /Users/testuser
name: testuser
passwd: *****
realname: TestUser
shell: /bin/tcsh
uid: 502
writing new directory /users/testuser
```

となる。

niutil コマンドを利用する niutil コマンドで新規ユーザを作成するには以下の手順を行えばよい。

```
niutil -create . /users/naito
niutil -append . uid 501
niutil -append . gid 20
niutil -append . home /Users/naito
niutil -append . shell /bin/tcsh
niutil -append . realname "Hisashi NAITO"
```

これは、新規ユーザの NetInfo エントリを作成し、順にデータを入力している。

【ユーザ情報の変更】 ユーザ情報を変更するには niutil コマンドを利用する。

ログインシェルを変更する 例えば、ユーザ情報のうち、ログインシェルを変更するには

```
sudo niutil -renameprop . /users/testuser shell /bin/bash
```

とすれば testuser のログインシェルが /bin/bash に変更される。

<sup>11</sup>【注】この \*\*\*\*\* の部分には、そのユーザの「暗号化されたパスワード」が書かれている。この「パスワードの暗号化文字列」を作成するには、以下のいずれかの方法を取るとよい。

- BSD システム (Mac OS X を含む) で crypt を使った簡単なプログラムを書いて、パスワードの暗号化文字列を生成する。(C のプログラムが書けるユーザなら、簡単に作成可能である。詳細は crypt(3) を参照。)
- Mac OS X 自身または、他の BSD システムから (例えば自分の) パスワードの暗号化文字列をコピーして、後にパスワードを変更する。
- この部分を空欄にしておき (つまり、「パスワードを空」にしておき) 後にパスワードを変更する。(この方法はあまりお勧めしない。)

なお、管理者が他のユーザのパスワードを変更するには、「システム環境設定」の「アカウント」で「パスワードのリセット」を行えばよい。また、passwd コマンドを passwd -i netinfo -l localhost/local username として利用する方法もある。



【ユーザの削除】 ユーザを削除するには `niutil` コマンドを利用するのがよい。

`testuser` を削除する

```
sudo niutil -destroy . /users/testuser
```

とすればよい。

### 18.2.3 トラブルの実例

この記事を書いているときに発生した重大なトラブルとその回復方法をレポートしておこう。実は `niload -v -m passwd` を実行するつもりで、誤って `niload -v -d passwd` を実行してしまった。つまり、既存のユーザデータベースを完全に削除してしまった。このとき筆者自身がどのようにこのトラブルから復帰したかを解説しておこう。

このトラブルはユーザデータベースを失ってしまい、ログインも `sudo` も何もできなくなったことが問題である。そこで、ユーザデータベースを回復するのが一番簡単で、`/var/db/netinfo` 以下のデータのバックアップが存在する場合には、それを何とかして書き戻せばよい<sup>12</sup>。ところが、そのバックアップは存在していなかったため、新規に NetInfo データベースを構築する必要があった。以下はそれを行う手順である<sup>13</sup>。

#### Single User Mode で起動する

この方法は [5, Section 15.6.6] を参照していただきたい。

ファイルシステムを書き込み可能にする

Single User Mode で起動すると、ファイルシステムは書き込み不可の状態では起動している。それを書き込み可能にするには

```
/sbin/fsck -fy  
/sbin/mount -uw /
```

を実行する。

#### NetInfo データベースの再構築と NetInfo の起動

既存の NetInfo データベースをバックアップしてから、データベースの再構築を行う。その後 NetInfo を起動する。

```
/bin/mv /var/db/netinfo/local.nidb /var/db/netinfo/local.nidb.old  
/usr/libexec/create_nidb  
/usr/sbin/nibindd &
```

を実行する。

#### NetInfo データベースでの情報の書き込み

再構築した NetInfo データベースには、筆者自身のユーザ情報は入っていないので、それを書き込む。

```
/usr/bin/nicl . -create /users/naito  
/usr/bin/nicl . -append uid 501  
/usr/bin/nicl . -append gid 20
```

<sup>12</sup>この場合には、“Single User Mode”で起動して、データの書き戻しを行い再起動すればよい。

<sup>13</sup>ただし、NetInfo データベースには「デフォルトから追加した情報」は「自分自身のユーザ情報だけ」という想定の話である。そうでない場合には、「追加されている情報」を復元する必要がある。とりあえず、自分自身がログインできて「管理者権限」が手にはいればよいという状況まで復活させるための方法である。

```
/usr/bin/nicl . -append home /Users/naito
/usr/bin/nicl . -append shell /bin/tcsh
/usr/bin/nicl . -append passwd ""
/usr/bin/nicl . -append realname "Hisashi NAITO"
/usr/bin/nicl . -insert /groups/admin users naito 2
```

を実行する<sup>14</sup>. この段階では「パスワード」は空にしておく. また, 「管理者」のリストに自分自身のユーザ名を付け加えておく.

#### パスワードの設定

パスワードは

```
/usr/bin/passwd
```

を実行して, インタラクティブに新規パスワードを設定する.

#### 再起動

これで一連の手順は終了で, データを保存して機器を再起動する.

```
/sbin/reboot
```

この後, 「アカウント」環境設定で「ログインアイコン」などを設定し直せばよい.

なお, NetInfo データベースの全データを取得するには

```
nidump -r / .
```

とすればよい. このデータがあれば niload を使ってデータベースを再構築することが可能になる.

## 18.3 その他のコマンド

その他の有用なコマンドとして, 次のものを挙げておこう.

- system\_profiler  
システム情報を出力する.
- defaults  
OS・アプリケーションなどのデフォルト設定を変更する.
- update\_prebinding  
システムライブラリなどの情報をアップデートする. 例えばアプリケーションの起動が遅くなったときなどに, 管理者モードで `update_prebinding -root /` とすると解消される場合がある.
- softwareupdate  
ソフトウェアアップデートの実行.

#### 【defaults の使用例】

ユーザ設定情報の取得

```
defaults read -globalDomain
```

---

<sup>14</sup>ここで, `-insert /groups/admin users naito 2` の “2” は, この `/groups/admin/users` のリストの 2 番目に `naito` を付け加えることを意味する.

とすると、ユーザ設定情報を得ることができる。

例えば、`defaults read -globalDomain AppleLanguages` とすると、そのユーザの言語環境設定のリストが出力される。

また、`defaults read com.apple.mail` とすると「Mail」アプリケーションの設定情報を得ることができる。ここで、各アプリケーションに対する「ドメイン名」(この例の `com.apple.mail` など) は、ライブラリフォルダ内の「Preferences」フォルダにある「それらしい名前」から知ることができる。または、`defaults domains` で全てのドメイン名を出力することもできる。

ユーザ設定情報の変更 例えば、そのユーザの言語環境設定のリストを変更するには、

```
defaults write -globalDomain AppleLanguages "(ja, en, fr)"
```

などとすればよい。この例では、「日本語、英語、フランス語」の順に言語環境の優先順位が設定される。

### 【softwareupdate の使用例】

利用可能なソフトウェアアップデート一覧の取得

```
softwareupdate -l
```

とすると、利用可能なソフトウェアアップデートの一覧を得ることができる。

```
# softwareupdate -l
Software Update Tool
Copyright 2002-2003 Apple Computer, Inc.

Software Update found the following new or updated software:
! iCal155-1.5.5
  iCal, 1.5.5, 7180K [required]
! MacOSXUpdateCombo10.3.9-10.3.9
  Mac OS X Update Combined, 10.3.9, 120092K [required]
  [restart]
! SecurityUpdate
  Security Update 2004-10-27, 1.0, 832K [required]
* AirPortSW-4.2
  AirPort Software, 4.2, 14652K [restart]
```

この中で！がついているものが、その機器でのアップデート対象のものとなる。

ソフトウェアアップデートの実行

ソフトウェアアップデートを実行するには、アップデートするものを指定してコマンドを実行する。

```
softwareupdate -i SecurityUpdate2004-10-27Pan-1.0 iCal155-1.5.5
```

とすると、指定した2つのソフトウェアアップデートを実行することになる<sup>15</sup>。

<sup>15</sup>ソフトウェアアップデートの実行には管理者権限が必要である。また、アップデート名はどのような順序で指定してもよい。

```
Software Update Tool
Copyright 2002-2003 Apple Computer, Inc.

Security Update: 0...10...20...30...40...
iCal: 0...10...20...30...40...
Security Update: 0...10...20...30...40...50...60...70...80...90.
..100
iCal: 0...10...20...30...40...50...60...70...80...90...100
Optimizing system performance. This may take a while...
Done.

You have installed one or more updates
that requires that you restart your computer.
Please restart immediately.
```

リストの中で `restart` となっているものをインストールした場合には、速やかな再起動を求められるので `reboot` コマンドにより再起動を行う。

## 19 起動の仕組み

最後に Mac OS X のシステムの起動手続きを調べておこう。

### 19.1 起動手続き

<sup>16</sup> 多くの UNIX システムの起動手続きは以下のような順序で行われる。(一番最初の部分はかなり省略してある。)

1. 電源が投入されるとファームウェアが起動し、ハードウェアのテストを行った後に起動デバイスのカーネルやハードウェア拡張を読み込み、ルートファイルシステムをマウントする。
2. `init` プロセスが起動して、`/etc/rc` などの起動スクリプトを読み出す。

ここで「起動スクリプトを読み出す」という部分はシステムによってかなり異なる。例えば FreeBSD の場合には `/etc/rc` を順次実行する中で `/etc/rc.conf`、`/etc/default/rc.conf` に記述されたホストごとの設定を評価する。Solaris の場合には「ランレベル」ごとに指定されたディレクトリ(例えば `/etc/rc2.d` など)の中のスクリプトを、スクリプトの名前の順序に従って実行する。いずれにしても、多くの UNIX システムでは各種のサービスを起動する順序は `/etc/rc` などに記述されていて、起動時に障害が発生した場合には、スクリプトを調べることによりどこで障害が発生しているかが容易に見えてくる。

Mac OS X の起動手続きは、これらのシステムとは全く異なる方法を取る。その手続きは以下の通りである。

1. 電源が投入されるとファームウェアが起動し、ハードウェアのテストを行った後に起動デバイスのカーネルやハードウェア拡張を読み込み、ルートファイルシステムをマウントする。
2. `init` プロセスが起動して、`/etc/rc` などの起動スクリプトを読み出す。`/etc/rc` では、仮想メモリを有効にした後に `SystemStarter` とよばれるプログラムを起動する。

<sup>16</sup>【注】名古屋大学情報連携基盤センターニュース [7] に記載したこの章の記述には少々誤りが含まれていたため、誤りを修正した。具体的な誤りの内容は以下の通りである。

「`/Library/StartupItems` にサードパーティ製サービスの起動スクリプトが置かれる」という記述は誤りである。Mac OS X 10.4 からは `/Library/StartupItems` に置いた起動スクリプトは、起動時には参照されなくなった。

3. SystemStarter は StartupItems ディレクトリにある起動スクリプトを、それぞれの依存関係を判断しながら「並行して」起動させていく。

他のシステムの起動手続きと「全く異なる」部分は、起動スクリプトを依存関係を判断しながら並行して起動していく部分である。StartupItems は /System/Library/StartupItems に保存されている。以下では /System/Library/StartupItems の内容とその役割を順に見ていこう<sup>17</sup>。

### 19.1.1 StarupItems の内容

デフォルトで /System/Library/StartupItems に含まれるディレクトリ（起動スクリプト）は以下にあげるものたちである。

- Apache: Apache Web サーバ。  
/etc/hostconfig に WEBSERVER=-YES- と記述されていれば apache を起動する。
- AppServices: アプリケーションサーバ。  
CoreService を起動する。
- AppleShare: AppleShare サーバ。  
/etc/hostconfig に AFPSEVER=-YES- と記述されていれば AppleShare サーバを起動する。
- AuthServer: Authentication サーバ。  
/etc/hostconfig に AUTHSERVER=-YES- と記述されていれば tim を起動する。
- CrashReporter: クラッシュレポートの起動。  
/etc/hostconfig に CRASHREPORTER=-YES- と記述されていれば crashreporterd を起動する。
- Disks: autodiskmount の実行。
- FiberChannel: Fiber Channel コントローラの設定を行う。  
Fiber Channel カードを搭載する場合のみ関係する。
- IFCStart: 国際化コンポーネントのキャッシュを再構成する。
- IPServices: InternetSharing の実行。/etc/com.apple.named.conf.proxy が存在するときのみ実行される。
- Metadata: メタデータサーバ (SpotLight のためのサーバ) の実行。  
/etc/hostconfig に SPOTLIGHT=-YES- と記述されていれば lsregister mds を起動する。
- NFS: NFS の設定。  
/etc/exports が存在すれば NFS サーバを起動し、/etc/hostconfig に AUTOMOUNT=-YES- と記述されていれば Directory Service から NFS ディスクの情報を得て automount を利用して NFS ディスクをマウントする。
- NIS: NIS の設定。  
/etc/hostconfig に NISDOMAIN が記述されていれば NIS を起動する。
- NetworkTime: NTPD を起動する。  
/etc/hostconfig に TIMESYNC=-YES- と記述されていれば, ntpd を起動する。

---

<sup>17</sup>/System/Library/StartupItems の内容は Tiger になって大幅に削減され、多くのものが /etc/rc 内で起動されるようになった。

- PrintingServices: プリンタサービスを起動する。  
/etc/hostconfig に CUPS=-YES- と記述されていれば, cupsd を起動する.
- RemoteDesktopAgent: Apple Remote Desktop クライアントの設定。  
/etc/hostconfig に ARDAGENT=-YES- と記述されているときのみ.
- SNMP: SNMP (Simple Network Managing Protocol) サーバ。  
/etc/hostconfig に SNMPSERVER=-YES- と記述されていれば snmpd を起動する.

これらのことからわかる通り, 多くのサービスは /etc/hostconfig に指定された変数に -YES-, AUTOMATIC+ などの値を与えることにより, サービスの起動を制御できる.

### 19.1.2 /etc/hostconfig のデフォルト設定

/etc/hostconfig のデフォルト設定は

```

HOSTNAME=-AUTOMATIC-
ROUTER=-AUTOMATIC-
AFPSSERVER=-NO-
APPLETALK=en0
AUTHSERVER=-NO-
AUTOMOUNT=-YES-
CONFIGSERVER=-NO-
CUPS=-YES-
IPFORWARDING=-NO-
IPV6=-YES-
MAILSERVER=-NO-
NETBOOTSERVER=-NO-
NETINFOSSERVER=-AUTOMATIC-
NISDOMAIN=-NO-
RPCSERVER=-AUTOMATIC-
TIMESYNC=-NO-
QTSSSERVER=-NO-
SSHSSERVER=-NO-
WEBSERVER=-NO-
SMBSSERVER=-NO-
DNSSSERVER=-NO-
CRASHREPORTER=-YES-
NFSLOCKS=-AUTOMATIC-
SNMPSSERVER=-NO-
SPOTLIGHT=-YES-
ENCRYPINGSWAP=-YES-

```

である. /etc/hostconfig の変数のいくつかは「システム環境設定」を通じて指定される.

- SSHSSERVER: 「システム環境設定」の「共有」設定で「リモートログイン」を有効にすると -YES- となる.
- WEBSERVER: 「システム環境設定」の「共有」設定で「パーソナル WEB 共有」を有効にすると -YES- となる.
- AFPSSERVER: 「システム環境設定」の「共有」設定で「パーソナルファイル共有」を有効にすると -YES- となる.
- SMBSSERVER: 「システム環境設定」の「共有」設定で「Windows ファイル共有」を有効にすると -YES- となる.
- TIMESYNC: 「システム環境設定」の「日付と時刻」設定で「ネットワークタイムサーバ」を有効にすると -YES- となる.

- ENCRYPTINGSWAP: 「システム環境設定」の「セキュリティ」設定で「安全な仮想メモリを使用」を有効にすると -YES- となる。

### 19.1.3 起動スクリプトの実行順序

これらの起動スクリプトは SystemStarter によって依存関係が判断されると書いたが、それぞれのスクリプトの依存関係は、各スクリプトが保存されているディレクトリ内に記述されている。各起動スクリプトのディレクトリ内には以下のファイル(ディレクトリ)がある。(これは /System/Library/StartupItems/AppleShare の例である。)

- AppleShare: 実行される起動スクリプト本体。
- StartupParameters.plist: 起動スクリプトの依存関係を記述したファイル。
- Resources: 各言語環境ごとに利用される「起動時のメッセージなど」を記述した xml 形式のファイルが格納されているディレクトリ。

Mac OS X でのスクリプト(アプリケーション)は単体で存在する場合もあるが、このようにディレクトリ全体で一つのスクリプト(アプリケーション)を構成している場合もあり、これを「アプリケーションフォルダ」と呼ぶ。

依存関係を記述したファイル(StartupParameters.plist)は AppleShare の場合、

```
{
  Description      = ``Apple File Service``;
  Provides         = (``Apple File Service``);
  Requires         = (``Disks``, ``DirectoryServices``, ``AppleTalk``);
  Uses             = (``Network Time``);
  OrderPreference = ``None``;
}
```

という内容であり、

- このスクリプトが Apple File Service という「識別子」であり(Provides の部分)。
- このスクリプトを実行するには Disks, DirectoryServices, AppleTalk を必要とし(Requires の部分)。
- このスクリプトを実行するには Network Time がもし存在すれば、その実行を必要とし(Uses の部分)。

ことが指定されている。また、OrderPreferences は、同じ属性を持つスクリプトが複数あった場合に、その中での順序を指定する役割を持ち、

- None: 特に順序は指定されない。
- First: 最も最初に行う。
- Early: 出来る限り早く実行する。
- Late: 出来る限り遅く実行する。
- Last: 最後に実行する。

という属性を指定できる。

したがって“Apple File Service”の実行は Disks, DirectoryServices, AppleTalk の起動が完了した後であり、Network Time が存在している場合には、その起動が完了した後に実行されることがわかる。



## 19.2 起動スクリプトの追加方法

システムの起動時に実行するサービスを追加するには  
/System/Library/StartupItems ディレクトリにスクリプトを追加すればよい。<sup>18</sup>  
起動スクリプトを作成するための手順は以下の通りである。以下では localconfig という名前の起動スクリプトを作る例を示す。

1. /System/Library/StartupItems に localconfig ディレクトリを作成する。
2. /System/Library/StartupItems/localconfig に StartupParameters.plist を作成する。その内容は以下のいずれかである。<sup>19</sup>

```
{
  Description      = "localconfig";
  Provides         = ("localconfig");
  Requires         = ("Network", "Network Time");
  OrderPreference = "None";
}
```

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist SYSTEM
  "file:///localhost/System/Library/DTDs/PropertyList.dtd">
<plist version="'0.9'">
<dict>
  <key>Description</key>
  <string>localconfig</string>
  <key>OrderPreference</key>
  <string>None</string>
  <key>Provides</key>
  <array>
    <string>localconfig</string>
  </array>
  <key>Requires</key>
  <array>
    <string>Network</string>
    <string>Network Time</string>
  </array>
</dict>
</plist>
```

前者は「古い形式」のものであり、最新のサービスは後者の「XML 形式」を取っているものが多い。

3. /System/Library/StartupItems/localconfig に実際の起動スクリプトである localconfig を作成する。通常このスクリプトは /bin/sh スクリプトであり、以下のように記述してあるものが多い。

<sup>18</sup>Version 10.3 以前の Mac OS X の基本コンセプトの下では /System ディレクトリ以下はデフォルトのシステムのみが使うことになっていたため、サードパーティ製のアプリケーションなどは /Library 以下の対応するディレクトリを利用すればよかった。しかし、Version 10.4 では /Library/StartupItems が参照されなくなったため、デフォルトのものとは後から追加したものの区別が困難になっている。

<sup>19</sup>この Requires, OrderPreference は一例であり、スクリプトの内容に依存する。



```
#!/bin/sh
. /etc/rc.common
StartService ()
{
    ##### ここにサービス起動のためのコマンド列を記述する。
    ConsoleMessage "Starting Local Configuration"
}
StopService ()
{
    ##### ここにサービス終了のためのコマンド列を記述する。
    ConsoleMessage "Stopping Local Configuration"
}
RestartService () { StopService; StartService; }
RunService "$1"
```

スクリプト内で `/etc/rc.common` を評価しているが、`/etc/rc.common` 内で `/etc/hostconfig` が評価されるので、このスクリプト内では `/etc/hostconfig` で指定した変数を参照可能となる。

このスクリプトは、システム起動時には `start`、システム終了時には `stop` を引数として実行され、システム起動時には、スクリプト内の `StartService` 関数が、システム終了時には、スクリプト内の `StopService` 関数が実行される。

4. 起動時に `/etc/rc` から呼び出される `SystemStarter` を利用して起動する場合に、コンソール（起動画面）にメッセージを表示したい場合には、さらに各言語ごとのリソースを作成する必要がある。

そのためには `/System/Library/StartupItems/localconfig` 内に `Resources` ディレクトリを作成し、その中に `English.lproj`（最低これだけは必要）ディレクトリを作成する。`English.lproj` ディレクトリ内には `Localizable.strings` ファイルを以下の形式で作成する。

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE plist SYSTEM
  "file:///localhost/System/Library/DTDs/PropertyList.dtd">
<plist version="'0.9'">
<dict>
    <key>Starting Local Congifuration</key>
    <string>Starting Local Configuration Script</string>
    <key>Stopping Local Configuration</key>
    <string>Stopping Local Configuration Script</string>
</dict>
</plist>
```

この XML ファイルは、上の `localconfig` スクリプト内で `ConsoleMessage` に指定した文字列を `key` として、`ConsoleMessage` のかわりに、その `key` に対応する `string` をコンソール（起動画面）に表示する意味を持つ。すなわち、`localconfig` 実行時にコンソールに表示されるメッセージは `Starting Local Configuration Script` となる。<sup>20</sup>

システムが「日本語」モードで動作する場合、`Japanese.lproj` ディレクトリ内に `Localizable.strings` が存在すれば、そこに記述された文字列がコンソールに表示される。なお、「日本語文字列」を表示させるためには `Localizable.strings` の `string` としては UTF-8 エンコーディングされた文字列を記述する必要がある。

<sup>20</sup>これは、もちろん Aqua 環境のコンソールの話である。

## 最後に

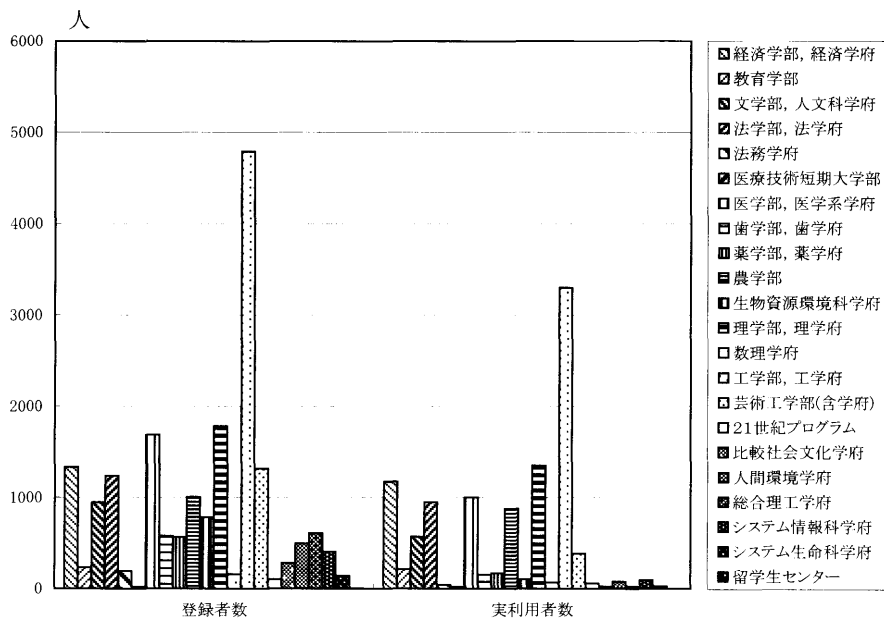
予定を大幅に超過して、さらには何度も原稿を落しながら、2年以上7回に渡って Mac OS X に関する解説を書いてきたが、一応今回を持って終りとしたい。この連載を通じて、Mac OS X のより進んだ利用法を理解して頂き、安全かつ快適な Mac Life を楽しんで頂ければ幸いである。最後に、この連載を楽しんで頂いた読者の方々に感謝したい。

## 参考文献

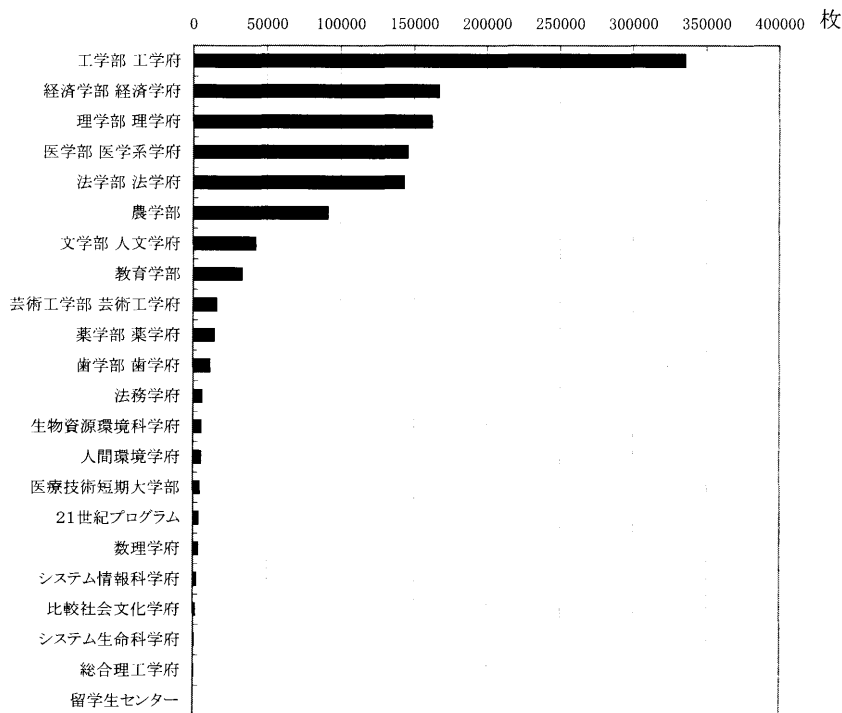
- [1] 内藤 久資, Mac OS X -先進的で直感的なオペレーティングシステム-.  
名古屋大学情報連携基盤センターニュース, 2 (2003), 201-245.
- [2] 内藤 久資, Mac OS X -あなたの Mac は元気ですか?-.  
名古屋大学情報連携基盤センターニュース, 2 (2003), 320-353.
- [3] 内藤 久資, Mac OS X -Mac OS X の進化論-.  
名古屋大学情報連携基盤センターニュース, 3 (2004), 9-34.
- [4] 内藤 久資, Mac OS X -Mac OS X のネットワーク-.  
名古屋大学情報連携基盤センターニュース, 3 (2004), 105-149.
- [5] 内藤 久資, Mac OS X -Mac OS X Server のススメ-.  
名古屋大学情報連携基盤センターニュース, 3 (2004), 290-317.
- [6] 内藤 久資, Mac OS X -続 Mac OS X の進化論-.  
名古屋大学情報連携基盤センターニュース, 4 (2005), 211-236.
- [7] 内藤 久資, Mac OS X -コマンドライン・システム管理入門-.  
名古屋大学情報連携基盤センターニュース, 5 (2006), 16-38.
- [8] Common UNIX Printing System, <http://www.cups.org/>
- [9] xinetd, <http://www.xinetd.org/>
- [10] The Apache Software Foundation, <http://www.apache.org/>

## 平成17年度教育用システム統計

### 1. 学部等別登録者数および実利用者数(平成17年4月～平成18年3月)



### 2. 学部・学府別のプリンタ出力状況(平成17年4月～平成18年3月)



## 人 事 異 動

## ◎転出・退職者等

発令年月日	異動後の職等	氏 名	異動区分	異動前の職等
H18.3.31		大西 政一	勸奨退職	システム運用係長
//	理化学研究所研究員	渡邊 秀和	退職	学術研究員
//	情報基盤センター学術研究員	本田 宏明	//	助手
//	研究部事務補佐員(パートタイム)	秋永 静枝	定年退職	研究部事務補佐員 (有期契約職員)
//		店田ひろみ	退職	研究部技術補佐員
//		黒木 圭容	//	庶務係事務補佐員
H18.4.1	理学部等用度係長	梅津 哲昭	配置換	会計係長
//	附属図書館図書館企画課 企画係長	瓜生 照久	//	電子図書館係長
//	工学部等総務課主任 (知能システム)	金島 晴子	//	庶務係主任

## ◎転入者等

発令年月日	異動後の職等	氏 名	異動区分	異動前の職等
H18.4.1	センター長(再任)	村上 和彰	職務附加	
//	副センター長(再任)	藤野 清次	//	
//	会計係長	橋本 幸男	配置換	農学部経理係長
//	電子図書館係長	青木 良秀	採用	有明高専庶務課図書係長
//	システム運用係長	堀 安一郎	//	北九州高専会計課専門職員
//	庶務係員	楠 美喜	配置換	附属図書館芸術工学分館情報 サービス係
//	研究部学術研究員(特任助教授)	顧 鳳龍	採用	
//	研究部学術研究員(特任助教授)	本田 宏明	//	
//	研究部学術研究員	小林 泰三	//	
//	研究部テクニカルスタッフ	首藤 里佳	//	
//	研究部テクニカルスタッフ	梶田めぐみ	//	
//	研究部事務補佐員(パートタイム)	秋永 静枝	//	
//	研究部技術補佐員(パートタイム)	西本麻美子	//	
//	庶務係事務補佐員(パートタイム)	杠 祐子	//	

新年度を迎えもうすぐ一月、ゴールデンウィーク間近のこの季節、何処を見ても色鮮やかな花々が咲き誇っています。真っ白い雪柳から桜、色とりどりのツツジ、藤、アメリカハナミズキ、牡丹、そして数々の草花。いろんな花が次々に咲いて明るい開放的な気分にしてくれます。どれも華やかで美しいのですがそんな中、私は、目立たずひっそりと優しく新緑の中に白く咲いているどうだんつつじに惹かれます。じっと眺めていると安らぎを覚えます。センター玄関周りのどうだんつつじも今が見頃です。人生も後半になるとこんな小さな花のすばらしさを痛感するものですね。

日記を書く時、最近では毎日のニュースに日本という国について考えさせられ、巷の出来事に一喜一憂 と言いたいのですがこの頃は喜ばしい事が余りにも少ないような・・・

でも自分自身のことになると話は別で、折に触れ周囲の方や子供達の思い遣りに喜びを感じ感謝しながらの毎日です。どうだんつつじを眺めながら私も同じねとついつい話しかけたくなります。小さくてもいいから優しい花を咲かせているそんな人生を最後まで送れたらいいな！と。

いつも小さなものに心惹かれる私ですが還暦の節目に大きな大きな自然に触れてこよとこのゴールデンウィークを利用して太古の時代から続く大陸の雄大な自然を見に出かけます。帰ってきてからの自分の内なる変化に期待しながら。

(また3年間勤めさせて頂くことになりましたS. A)

つい先日、遅まきながら私も iPod (nano 2GB)ユーザーになってしまいました。iPod もいろんな種類が出ていますし、最近では他社製もたくさんあるので迷いましたが、薄さ・手軽さ・人気、と予算で選びました。同じくらいの性能のものを使ったことがないので、今まで使っていたMDプレイヤーとしか比較できませんが、とっても良い！どうしてもっと早く買わなかったんだ！！控えめに 2GB のものにしたので収録できる曲数は約 500 曲ですが、それでも充分楽しめます。今までは持ち運びが面倒だったので、MD一枚を入れればなしにして飽きるほど聴いていましたが、500 曲もあればいろんなジャンルの曲を楽しめて、それだけで大満足です。CDからの取込みも専用ソフトの操作が簡単なので、何枚ものCDを取り込む作業も全然苦になりませんでした。ほんとに画像も収録できたり、まだまだ便利機能があるのだと思いますが、Windows しか使ったことがない私はクリックホイールの使い方がわからず悪戦苦闘したくらいですから、ちょっとずつ楽しんでいこうと思っています。皆さんも快適オーディオ生活を手に入れませんか？(決して回し者ではありません)

(次は dyson の掃除機が気になる N)

## 投稿のしおり

九州大学情報基盤センターでは、利用者の方々との深い交流と有益な情報交換のため、以下の種類の原稿を募集しています。

- 随想
- 計算機を利用した研究・開発の紹介
- アプリケーションの実用例や解説
- プログラミングの実例や解説
- センターに対する質問・要望
- 利用者の声
- 計算機やネットワークに関すること

執筆の際は、後述の《執筆上の注意》を必ずご参照ください。投稿された原稿の掲載については、広報委員会で検討させていただきます。

別刷りは、原稿提出時に希望があれば、50部まで無料で差し上げます。50部以上希望される場合は、原稿提出時にご相談ください。

なお、投稿に関するお問合せは共同利用係(ダイヤルイン092-642-2305)までお願いします。

### 《原稿送付先》

原稿の送付先は、以下のとおりです。

〒812-8581  
福岡市東区箱崎6丁目10番1号  
九州大学 情報基盤センター  
共同利用係 行

### 《メ切》

原稿のメ切は、5・9・1月の15日です。発行は、その2ヶ月後になります。メ切を過ぎた原稿は、その次の号への投稿扱いとなる場合があります。

### 《執筆上の注意》

1. 原稿は原則としてワードプロセッサ等の出力結果とします。

用紙サイズはB5のみとし、書式は図1をご覧ください。手書きでも結構ですが、その場合センター規定の原稿用紙を使用してください。用紙は、共同利用係にあります。フロッピーディスクや電子メールでの投稿は別途、共同利用係までご相談ください。

2. ワードプロセッサによる出力は、写真製版とさせていただきますので、投稿原稿の品質にはご注意ください。
3. 手書きの場合は、黒鉛筆、黒ボールペンまたは黒インクのいずれかで書いてください。また、数字、英文字、大(小)文字、上(下)付き文字等混合しやすい文字ははっきりわかりやすく指定してください。
4. 原則として、常用漢字、現代かな使いで統一してください。
5. 第一ページには、必ず題名、著者名、所属および電子メールアドレスを記入してください。記入位置については、図1をご覧ください。

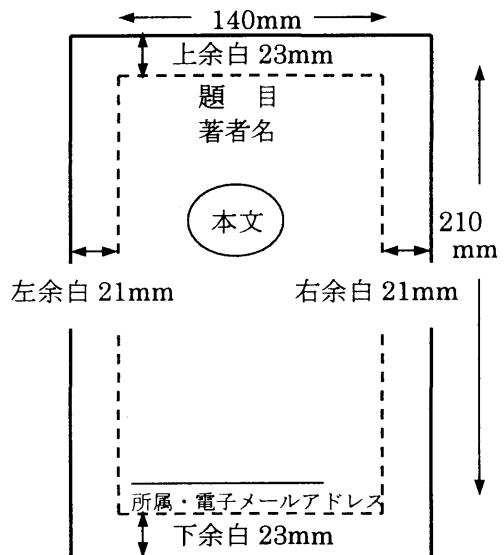


図1：書式設定

# 全国共同利用版 Vol.6 No.1 (2006年5月発行) の紹介

本号より、裏表紙に本センター全国共同利用版広報誌の目次を掲載しております。これは、主に全国共同利用施設として運用している研究用計算機システムの利用者向けに発行している広報誌です。

本誌(学内共同利用版)を含め、これらの広報記事は以下のページより閲覧して頂けますので、興味のある記事がありましたらアクセスして下さい。

<http://www.cc.kyushu-u.ac.jp/publish/kohobkno/index.html>

今回の内容は以下の通りです。

## ● 解説

- Orphan 指示文を用いた粗粒子 OpenMP 並列プログラム
- 分子科学計算推進室の設置のお知らせ
- 分子科学計算推進室からのお便り

## 全国共同利用版 Vol.6 No.1 (2006年5月発行) の紹介

九州大学情報基盤センターでは、本誌の他に全国共同利用施設として運用している研究用計算機システムの利用者向けに全国共同利用版広報誌を発行しています。今回の内容は以下の通りです。

### ・ 解説

- Orphan指示文を用いた粗粒子 OpenMP並列プログラム
- 分子科学計算推進室の設置のお知らせ
- 分子科学計算推進室からのお便り

これらの広報記事は以下のページより閲覧して頂けます。

<http://www.cc.kyushu-u.ac.jp/publish/kohobkno/index.html>

九州大学情報基盤センター広報

Vol. 6, No. 1

平成18年5月 発行

編集 九州大学情報基盤センター

広報委員会

印刷 アイメディア株式会社