

SQLを用いた日経総合経済ファイル 読出しシステムの開発*

Constructing a Database System for Nikkei Sogo Keizai File using SQL

豊原法彦

We developed a sql based data retrieval system. In this system, all field names are also databased, so a user scribes only to a specific sql-code to get data without referring any other field name in the same code book. We used a relational data base system, PostgreSQL version 8.4 on linux, which is distributed as UBUNTU 9.04. In this paper, we introduce a way to develop this system and show some examples to retrieve them.

Norihiko Toyohara

JEL : C82, C87

キーワード : 日経総合経済ファイル データベース構築 データベース管理

Key words : Nikkei sogo keizai file, Constructing Data Base, Data Base Management

1 はじめに

web2.0などと称されるインターネットを通じた情報環境の進化に伴って、データ抽出作業の可用性向上、シームレス化とメンテナンスの簡便性が求められている。本学産業研究所とともに開発したマクロデータの読み出しシステム¹⁾に関しても、同様のコンセプトに従ってSQL²⁾言語を使用して活用できる

* 本論文は 2008 年度関西学院大学個人特別研究費の 1 成果である。

1) KG-edens; KwansaiGakuin Economic Data Extraction Network System。詳しくは豊原 (1999) を参照のこと

2) 一般に、structured query language の略と言われる

ものを設計、構築した。このことによって、あらかじめ SQL 文を作成すれば、一括した抽出作業を行うことができることになった。

また、前回採用した web によるアクセス、学内での利用に制限するというコンセプトは踏襲し、ユーザのローカル環境にはできる限り依存しない形とした。このため、本システムでは、開発環境が安価で構築できる linux³⁾をベースに、データベースサーバ⁴⁾を用いた。なお、データアクセスについては基本的に学内からの機器に限定⁵⁾しており、サーバは静的プライベートアドレスを割り当てられている。

以下、2 章では最初に日経総合経済ファイルのデータ構造と特徴、3 章では具体的なデータベース作成、更新手順、4 章では利用するための方法、特に SQL 文について述べる。そして最後に今後の情報環境の変化に伴うデータベースと計量統計分析の関わりについてまとめる。

2 日経総合経済ファイルのデータ構造と特徴

ここで扱う日経総合経済ファイル⁶⁾は、同社の web サイト⁷⁾にあるように、日本経済に関わる国民所得統計、生産活動、企業経営、金融・財政、貿易・国際収支など国内の主要経済統計などのデータが 1 枚の cd-rom に、データ部分⁸⁾と属性部分⁹⁾に分けられて収録されている。

データ部分は表 1 のように 1 レコード 70 バイトの固定長形式で 6,193,721 レコード収録されている。各フィールドは順にレコード ID(2 ケタ; N0 ならばコード, M0 ならばデータを示す)、識別フラグ (1 ケタ; 本データベースの場合は 1 に固定)、系列コード番号 (24 ケタ; 左詰め)、期種コード (2 ケタ; 左詰め; A

3) UBUNTU 9.04 詳しくは <http://www.ubuntu.jp/>を参照

4) Postgres バージョン 8.4。詳しくは、<http://www.postgresql.org/>を参照

5) リモート p c 機能などを用いる際には、この限りではない

6) 本学産業研究所が日本経済新聞デジタルメディア社より購入

7) <http://www.nikkei.co.jp/needs/contents/macro.html>

8) CENTADATA. +日付 (YYYYMMDD 形式) (年を 4 ケタ, 月を 01 から 12, 日を 01 から 31 で表した形式 以下同じ)

9) CENTDOCA. +日付 (YYYYMMDD 形式), なお日本語は JIS X 0208 形式 (いわゆる Shift_JIS)

表 1: CD-ROM で提供されるフォーマット (データ部分)

0	1	2	3	4	5	7
1234567890123456789012345678901234567890123456789012345678901234567890123456789012						
M01ABDTFB	-----	M	_00321965010120010301			
M01ABDTFB	-----	M	_19650101	-----	2705.130000	
M01ABDTFB	-----	M	_19650201	-----	2705.170000	
M01ABDTFB	-----	M	_19650301	-----	2660.950000	
M01ABDTFB	-----	M	_19650401	-----	2451.020000	
M01ABDTFB	-----	M	_19650501	-----	2443.060000	
M01ABDTFB	-----	M	_19650601	-----	2434.280000	
M01ABDTFB	-----	M	_19650701	-----	2237.640000	
M01ABDTFB	-----	M	_19650801	-----	2218.830000	
M01ABDTFB	-----	M	_19650901	-----	2210.800000	
		(中略)				
M01ZS11	-----	Q	_20060701	-----	3.500000	
M01ZS11	-----	Q	_20061001	-----	3.700000	
M01ZS11	-----	Q	_20070101	-----	4.100000	
M01ZS11	-----	Q	_20070401	-----	3.700000	
M01ZS11	-----	Q	_20070701	-----	3.500000	
M01ZS11	-----	Q	_20071001	-----	3.400000	
M01ZS11	-----	Q	_20080101	-----	3.600000	
M01ZS11	-----	Q	_20080401	-----	3.500000	
M01ZS11	-----	Q	_20080701	-----	2.700000	
M01ZS11	-----	Q	_20081001	-----	1.700000	

(年次), SA (半年), Q (四半期), M (月次), データ日付 (8 ケタ; YYYMMDD 形式), データ値 (20 ケタ; 小数点以下 5 桁), 速報フラグ (1 ケタ; 速報データの場合 1, それ以外は 0), 予備 (12 ケタ) である。

これはたとえば表 2 といった perl のプログラムを実行することによってデータ部分だけを区切り子を tab¹⁰⁾で出力できる。なおここでは固定長の部分を読み込むだけでなく、あわせて冗長なスペースコードも取り除いている。具体的には入出力環境を設定した後に、1 行全体 (70 文字 + 2 文字の改行コード¹¹⁾) を読み込み、substr コマンドにて適切な長さに分割、格納後、冗長な空白を乗り除き、区切り子を tab として出力するという処理をファイル終端まで繰り返す。

次に、データ属性に関するファイルについて述べる。CD-ROM に収録され

10) HEX コード 09

11) Carriage Return(CR) と Line Feed(LF)

表 2: CD-ROM からデータを読み出し可変長で保存するスクリプト (perl)

```

open (INO,"<入力ファイル");
open (OUT,">出力ファイル");
$init=0;
do {
  read (INO,$xx,72);

  $a[1]=substr($xx, 0, 2);
  $a[2]=substr($xx, 2, 1);
  $a[3]=substr($xx, 3, 24);
  $a[4]=substr($xx, 27, 2);
  $a[5]=substr($xx, 29, 8);
  $a[6]=substr($xx, 37, 20);
  $a[7]=substr($xx, 57, 1);
  $a[8]=substr($xx, 58, 12);

  if ($a[1] ne "N0") {
    $a[1] =~ s/\x20//g;$a[2] =~ s/\x20//g;$a[3] =~ s/\x20//g;$a[4] =~ s/\x20//g;
    $a[5] =~ s/\x20//g;$a[6] =~ s/\x20//g;$a[7] =~ s/\x20//g;$a[8] =~ s/\x20//g;
    print OUT "$a[1]\t$a[2]\t$a[3]\t$a[4]\t$a[5]\t$a[6]\t$a[7]\t$a[8]\n";
  }
} while (eof(INO)!=1);

```

表 3: CD-ROM で提供されるフォーマット (属性部分)

```

N11ABDTFB-----M_ASSETS_OF_TRUST_FUND_BUREAU-MINISTRY_OF_FINANCE-BANK_DEBENTURES-----
-----100 MILLION_YEN-----
-----FINANCE_AND_ECONOMIC_STATISTICS_MONTHL
Y-----
N21ABDTFB-----M_資金状況_資金運用部_金融債-----
-----億円-----
-----金融経済統計月報-----
N11ABJAD-----M_ACCOUNTS_OF_THE_BANK_OF_JAPAN-ASSETS-OTHER_SECURITIES_OUTSTANDING-----
-----100 MILLION_YEN-----
-----FINANCE_AND_ECONOMIC_STATISTICS_MONTHL
Y-----
N21ABJAD-----M_目録勘定_目録勘定_その他債券-----
-----億円-----
-----金融経済統計月報-----
(中略)
N11ZS10-----Q_RATIO_OF_CURRENT_PROFITS_TO_SALES-ALL_INDUSTRIES-----
-----QUARTERLY_REPORTS_OF_INCORPORATED_ENT
ERPRISES_STATISTICS-----
N21ZS10-----Q_法人企業_売上高経常利益率(全産業)-----
-----パーセント-----
-----法人企業統計季報-----
N11ZS11-----Q_RATIO_OF_OPERATING_PROFITS_TO_SALES-ALL_INDUSTRIES-----
-----QUARTERLY_REPORTS_OF_INCORPORATED_ENT
ERPRISES_STATISTICS-----
N21ZS11-----Q_法人企業_売上高営業利益率-----
-----パーセント-----
-----法人企業統計季報-----

```

ている形式は表 3 にあるように固定長 (500 バイト) である。なおこの表において英文行と日本文行でカラムがそろっていないように見えるのは、文字間サイズと改行位置の違いによるものである。このファイルレイアウトは、1 つのデータに対して英文、日本文の 2 つのレコードが割り当てられ、各レコードにある 8 つのフィールドは順に英日識別フラグ (1 ケタ; 最初左から 2 文字目が 1 の場合は英文, 2 の場合は日本文), 識別フラグ (1 ケタ; 本データの場合は 1 で固定), 系列コード番号 (24 ケタ;), 期種コード (2 ケタ;), 英文 (また日本文) 系列名称名 (254 ケタ; 左詰め), 英文 (また日本文) 単位名 (80 ケタ; 左詰め), 英文 (また日本文) 出典名 (80 ケタ; 左詰め), 予備 (57 ケタ) である。これを読み出すための perl によるスクリプトは、たとえば表 4 のような perl のプログラムであり、先の場合と同様に、ファイル設定を行った後に、1 行全体 (500 バイト + 2 文字の改行コード) を読み込み、先のバイト数に分割し、空白削除したものを、こちらの場合は” | ” (半角パイプライン) を区切り子として出力している。これは文字コードによっては日本語表記の際に tab キー (x09) に相当するコードが利用されているリスクを避けるためである。なお、一般に

表 4: CD-ROM からデータ属性を読み出し可変長で保存するスクリプト (perl)

```

open (INO,"<入力ファイル");
open (OUT,">出力ファイル");
$init=0;
do {
  read (INO,$xx,502);
  $a[1]=substr($xx, 0, 2);
  $a[2]=substr($xx, 2, 1);
  $a[3]=substr($xx, 3, 24);
  $a[4]=substr($xx, 27, 2);
  $a[5]=substr($xx, 29,254);
  $a[6]=substr($xx,283, 80);
  $a[7]=substr($xx,363, 80);
  $a[8]=substr($xx,443, 57);# 予備
  for ($i=1;$i<=8;$i++){
    $a[$i] =~ s/\x20//g;;
    print OUT "$a[$i]\|";
  };
  print OUT "\n";
} while (eof(INO)!=1);

```

データ部分、属性部分を出力する際には/tmp ディレクトリーを用いている。

3 データベースの構築手順

データベース構築の手順は、データ部分、属性部分のいずれも大きく 2 段階からなり、第 1 段階ではデータの整形、第 2 段階ではデータベースへの読み込みと利用可能な環境の構築である。

具体的には、perl によるプログラム表 2 を用いてデータを tab 区切りの可変長形式に加工し、また冗長な空白（半角全角のいずれも）をできる限り削除する。これによって、ファイル容量の縮小だけでなく検索および抽出を高速化するインデックスファイルを作成する際の作業時間および冗長ではない項目名を指定できることで実際の検索時間短縮効果も期待できる。これによって、表 5 のようなファイルが生成される。このファイルの各フィールド長は可変であり、tab によって区切られている。この処理によってファイルサイズは固定長の約 52%¹²⁾となった。

表 5: データベース読み込み用に加工されたファイル (\t は tab コード)

```

M01ABDTFB \t M \t 19650101 \t 2705.130000
M01ABDTFB \t M \t 19650201 \t 2705.170000
M01ABDTFB \t M \t 19650301 \t 2660.950000
M01ABDTFB \t M \t 19650401 \t 2451.020000
M01ABDTFB \t M \t 19650501 \t 2443.060000
M01ABDTFB \t M \t 19650601 \t 2434.280000
M01ABDTFB \t M \t 19650701 \t 2237.640000
M01ABDTFB \t M \t 19650801 \t 2218.830000
M01ABDTFB \t M \t 19650901 \t 2210.800000
(中略)
M01ZS11 \t Q \t 20060701 \t 3.500000
M01ZS11 \t Q \t 20061001 \t 3.700000
M01ZS11 \t Q \t 20070101 \t 4.100000
M01ZS11 \t Q \t 20070401 \t 3.700000
M01ZS11 \t Q \t 20070701 \t 3.500000
M01ZS11 \t Q \t 20071001 \t 3.400000
M01ZS11 \t Q \t 20080101 \t 3.600000
M01ZS11 \t Q \t 20080401 \t 3.500000
M01ZS11 \t Q \t 20080701 \t 2.700000
M01ZS11 \t Q \t 20081001 \t 1.700000
    
```

12) 30226981 バイト / 58309560 バイト = 51.83%

第2段階ではそのファイルをデータベースで利用するためにデータベースソフト上でSQL文とデータベースコマンドを実行する。つまり、データベースの定義¹³⁾、データ複写¹⁴⁾、インデックス作成¹⁵⁾、公開許可手続き¹⁶⁾を行う。このためのスクリプトが表6である。これで、データベース上に edens09 というデータベースが構築され、インデックスとして項目コードが設定される。

次に、属性ファイルのデータベース化を行う。データ部分と同様に、具体的には、perlによるプログラム表4を用いてデータを「|」（パイプライン）区切りの可変長形式に加工し、また冗長な空白（半角全角のいずれも）をできる限り削除する。さらにこのファイルは英文と日本語が混在しているので、日本語コードをlinux上で扱いやすいユニコード¹⁷⁾に変換する¹⁸⁾。その結果が表7である。なおこの表でも区切り子は「|」である。

そして、その加工を行ったファイルに対して、表8のスクリプトを実行する

表 6: データ部分をデータベースに読み込むためのスクリプト (sql)

```
create table データベース名 (  
  dum1 text,  
  dum2 text,  
  item text,  
  kind text,  
  date text,  
  value text,  
  quick_data text,  
  dum8 text  
);  
copy edens08 from 'tab 区切りされたデータファイル' delimiter E'\|';  
create index edens09_itemcode on edens09(item);  
GRANT SELECT ON edens08 to public;
```

13) create table コマンド

14) copy コマンド

15) create index コマンド

16) grant select コマンド

17) utf-8。Unicode Transformation Format-8

18) ここでは “ nkf -w < 入力ファイル > 出力ファイル ” コマンドを用いた

表 7: | (パイプ) で区切られた属性部分 (余分なスペース削除済み)

```
N11|ABDTFB|M |ASSETS OF TRUST FUND BUREAU-MINISTRY OF FINANCE-BANK DEBENTURES |100 MILLION YEN |F
INANCE AND ECONOMIC STATISTICS MONTHLY||
N211|ABDTFB|M |資金状況 資金運用部 金融債 |億円 |金融経済統計月報 ||
N11|ABJAD|M |ACCOUNTS OF THE BANK OF JAPAN-ASSETS-OTHER SECURITIES OUTSTANDING|100 MILLION YEN |FI
NANCE AND ECONOMIC STATISTICS MONTHLY||
N211|ABJAD|M |日銀勘定 日銀勘定 その他債券 |億円 |金融経済統計月報 ||
(中略)
N11|ZS10|Q |RATIO OF CURRENT PROFITS TO SALES-ALL INDUSTRIES |% |QUARTERLY REPORTS OF INCORPORATED
ENTERPRISES STATISTICS||
N211|ZS10|Q |法人企業 売上高経常利益率 (全産業) |パーセント |法人企業統計季報 ||
N11|ZS11|Q |RATIO OF OPERATING PROFITS TO SALES-ALL INDUSTRIES|% |QUARTERLY REPORTS OF INCORPORATED
ENTERPRISES STATISTICS||
N211|ZS11|Q |法人企業 売上高営業利益率|パーセント |法人企業統計季報 ||
```

表 8: 一次加工されたデータ属性ファイルをデータベースに読み込むためのスクリプト (SQL)

```
drop table edens09_header;
create table edens09_header(
    dum1 text,
    dum2 text,
    itemcode text,
    kind text,
    name text,
    unit text,
    source text,
    dum7 text,
    dum8 text
);
copy edens09_header from '/tmp/edens09_header_utf8.csv' delimiters E'|';
GRANT SELECT ON edens09_header to public;
```

ことで属性ファイルを edens09_header というファイル名でデータベースに登録する。併せて公開のための設定も行われている。

4 利用の方法

実際にデータを読み出すためには、次の 2 つの手順が必要である。第 1 段階はブラウザーによるサーバへのアクセスであり、第 2 段階は SQL 文の入力である。データ購入時の契約上利用が学内に限定している本システムでは、サーバにはプライベートエリア内での静的アドレスが設定されているので学外から

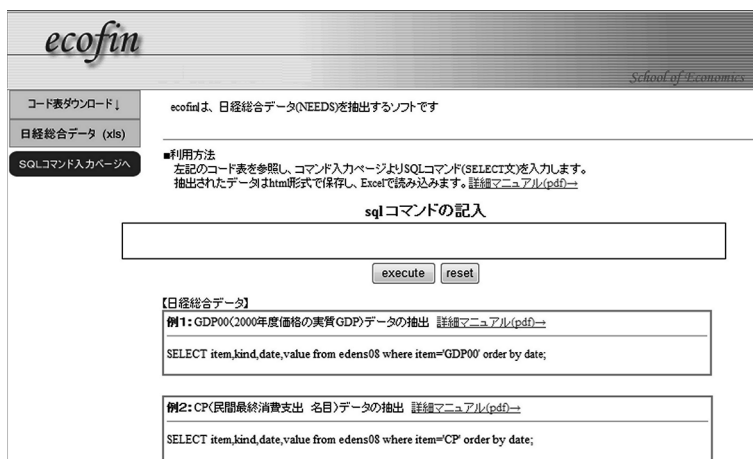


図 1: 初期画面

の利用はできないが、ユーザサイドにあらかじめ特定のソフトやドライバーなどの設定なしに読み出しができる。そのために、サーバサイドでは html¹⁹⁾ 文で cgi²⁰⁾ を使い、postgres に SQL 文を受け渡す²¹⁾ という手続きを行う。また抽出結果は html 文で出力される²²⁾ ので、ユーザがその結果をブラウザ上でその形式にて保存し、表計算ソフト²³⁾ でそのファイルを開くことによって利用することができる。

具体的には、ユーザが当該サーバにブラウザでアクセスすると、図 1 の画面が表示される。ここではコード表、使用例と並んで、実際に SQL 文を入力する部分が表示される。ユーザは抽出したいデータのコード番号をあらかじめ調べておく必要があるが、コード表をダウンロードして表計算ソフトに読み込みそのフィルタ機能やデータベース機能を用いる。または後に述べるようにデー

19) hyper text markup language

20) common gateway interface

21) " psql -f 一時保存ファイル名" コマンドを利用

22) psql -h コマンドを利用

23) たとえばマイクロソフト社製 excel など

データベース化されているコード表から直接検索することもできる。

以下ではまず、属性部分から作成したヘッダファイルにおいて「GDP」という文字列を項目名に含む項目の一覧を表示させる。

```
SELECT * from edens09_header where itemcode like '%GDP%';
```

なお、%はいわゆるワイルドカードで、空白を含む任意の長さの文字に対応する。このコマンドの結果が表 9 である。もちろん、日本語だけを表示することも可能である²⁴⁾。

これらの項目の中から「GDP」の 2000 年以降を取り出すコマンドは次の通りであり、これを実行すると、表 10 が得られる。

```
SELECT date,value from edens09 where item = 'GDP' and date >='20000101';
```

これらのデータを活用するには、ブラウザ上でこれが表示された後、表示されている必要な部分をコピーし表計算ソフトにペーストすることもできるが、

表 9: GDP を含む項目名一覧

```
ecofin=# SELECT * from edens09_header where itemcode like '%GDP%';
dum1 | dum2 | itemcode | kind |
      |      |          |      |
      |      | name     |      |      | unit
      |      |          |      |      |
      |      |          |      |      |      | source
N1   | 1   | GDP     | Q   | GROSS DOMESTIC PRODUCT(OR GROSS DOMESTIC
EXPENDITURE) | BILLION YEN
      |      |          |      | ANNUAL REPORT ON NATIONAL ACCOUNTS
N2   | 1   | GDP     | Q   | SNA 統合勘定 国内総生産・総支出
      |      |          |      | | 10億円
      |      |          |      | 国民経済計算年報
N1   | 1   | GDP&   | Q   | GROSS DOMESTIC PRODUCT(OR GROSS DOMESTIC
EXPENDITURE) | BILLION YEN
      |      |          |      | ANNUAL REPORT ON NATIONAL ACCOUNTS
N2   | 1   | GDP&   | Q   | SNA統合 SNA 国内総生産・総支出
      |      |          |      | | 10億円
      |      |          |      | 国民経済計算年報
      |      |          |      | (中略)
N1   | 1   | RESGDP00FYBY95 | A   | GDP RESIDUAL(CHAIN-LINKED)-FISCAL YEAR
      |      |          |      | | BILLION YEN
      |      |          |      | ANNUAL REPORT ON NATIONAL ACCOUNTS
N2   | 1   | RESGDP00FYBY95 | A   | SNA 主要系列表 (実質・連鎖価格、原数
      |      |          |      | 値) 開差 年度 | 10億円
      |      |          |      | 国民経済計算年報
(146 rows)
```

24) SELECT * from edens09_header where itemcode like '%GDP%' and dum1='N2';

ファイルとして保存して表計算ソフトに読み込むことも可能である。

次に本システムの動作プロセスについて述べる。ユーザが入力した SQL 文は、表 11 に引き継がれる。そこでは読み込んだ文字列を再構成し、システム

表 10: GDP の 2000 年以降抽出結果

date	value
20000101	122231.20000
20000401	124497.00000
(中略)	
20080701	123411.40000
20081001	130256.00000
20090101	115535.30000

(37 rows)

表 11: psql を呼び出すための cgi

```
#!/usr/bin/perl -U

read(STDIN,$buffer,$ENV{'CONTENT_LENGTH'});
@aa = split(/\+/, $buffer);
@aa0 = split(/=/, @aa[0]);
open (OUT1,"> /tmp/aa$$$.sql");
$ab=@aa0[1];
for($i=1;$i<@aa;$i++){
    $ab=$ab." ". @aa[$i];
}
$ab =~ s /'%'3D' /=/;
$ab =~ s /"%22"/\"/;
$ab=~ s/%([a-fA-F0-9][a-fA-F0-9])/pack("C",hex($1))/eg;

print OUT1 "$ab";
close (OUT1);

print "Content-type: text/html\n\n";
print "<HTML><BODY>\n\n";
# print $buffer;
print $ab;

a<-system("psql ecofin -l -Uecofin01 -f /tmp/aa$$$.sql -H ");
print a;
print "</BODY></HTML>\n\n";
```

上の予約語や数字、特殊記号と見なされるものをすべてプレーンな文字列に変換し、tmp にプロセス id²⁵⁾を付加して保存する。そし web 形式に出力する設定を整えた後、先のファイルを psql コマンドの引数として実行し、結果を html 出力する。

5 今後の課題

日経総合経済ファイルは汎用ホストコンピュータの時代に MT²⁶⁾で発売されて以来の歴史があるが、メディアの変化だけでなく、収録フォーマットもいわゆる 2000 年問題対応を経て現行形式の 1 レコード 1 データ型へと変化している。これはサーバの記憶容量が格段に増加するにつれてデータの可搬性よりも可用性が優先されたためであると考えられる。それに伴って読み出すための環境も再構築したものが本システム²⁷⁾である。

今後の情報環境の進展、特にサーバに資源を集中させ、クライアントには最小限のストレージと入出力を担わせる、いわゆるシンクライアント²⁸⁾化を考えると、経済分析を行うための手順もそれに併せて変化していくことになる。たとえば、ユーザがデータバンクからあらかじめ読み出したものを逐次加工しながら処理していくというこれまでのプロセスが、あたかもデータ分析ソフトにデータバンクが付属しているかのようにシームレスなシステム設計が開発されねばならない。また、教育面でも、計量的手法が進化し一層細分化されるにつれて、精緻な手法を入門段階から求められるようになる。さらに計量理論の側面も、シミュレーションを通じた理解も必要となろう。これらの点を考えるとき、より柔軟なユーザインターフェイスをもったシステム構築もある程度考えねばならない。

25) システム上から各プログラムに与えられた識別番号。\$\$で取得。

26) 磁気テープ

27) 従前のももメンテナンスされている

28) thin client

参考文献

豊原法彦, 「KG-EDENS; 関西学院大学経済データ抽出システムの構築」, 『経済学論究』第 53 巻第 3 号 (1999 年 12 月)。

< perl に関するもの >

Nathan Patwardhan, Ellen Siever et al., Perl リソースキット UNIX 版, オライリー・ジャパン, 1998/11。

Ellen Siever, Stephen Spainhour, Nathan Patwardhan (近藤嘉雪監訳), Perl クイックリファレンス, オライリー・ジャパン, 2000/3。

Shawn P.Wallace (田中幸訳), Perl/GNU ソフトウェアによる Web グラフィックスプログラミング, オーム社, 2000/08。

Larry Wall, Jon Orwant, Tom Christiansen (近藤 嘉雪訳), プログラミング Perl (VOLUME1), オライリー・ジャパン, 2002/09。

< PostgreSQL に関するもの >

石井 達夫 監修、浅羽 義之 他著, PostgreSQL 徹底入門 第 2 版, 翔泳社, 2008/04。