

地域総合データ読出しシステムの構築*

ToRADES; Total Regional Area Data Extraction System

豊原法彦

The ToRADES (Total Regional Area Data Extraction System) is a data retrieval system for some regional area data such as the population, budget, and product for each prefecture, city, ward, and village. As the number of each regional agent is more than 3000, and the number of item is beyond 2000, we use a relational data base system, PostgreSQL version 7.2 on linux, through ODBC to Microsoft Excel et.al. In this paper, we introduce a way to develop this system and show some examples to retrieve data.

Norihiko Toyohara

JEL : C82, C87

キーワード : 地域データ、データベース構築、データベース管理

Key words : Area Data, Constructing Data Base, Data Base Management

1 はじめに

これまでに本学で産業研究所とともにわれわれが開発してきたマクロデータの読み出しシステム¹⁾は、学内からのアクセスを前提に web よる利用が中心となっていた。この方法では、学内にあれば OS を問わずブラウザさえ利用可能であればデータ読み出しできるというメリットがある反面、反復利用が多い研究者にとっては、サーバにログインしてコマンドラインから利用できるモードを別途備えているとはいえ、取り出し手順やインターフェイスに煩瑣なところ

* 本システムの開発には 2002 年度関西学院大学研究設備計画から助成を受けた。記して謝意を表す。

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

ろがあったことも否めなかった。

それに対して今回の対象としているデータは、

1) その量は約 2.4 ギガバイトと、マクロデータ (60 メガバイト) とは比較にならないほど膨大 (いずれも非圧縮時)。

2) 抽出条件は都道府県市区町村, 項目, 年度とカテゴリーが広い。

という違いがある。従って, マクロデータの場合であればメンテナンスやポータビリティ重視して, 分割型のシーケンシャルファイルをデータベースとして利用することによって十分なアクセスを維持できた²⁾のに対して, 地域データの場合には同様の方法では十分なパフォーマンスが得られないことが明らかとなってきた。また, 抽出条件についても, たとえば兵庫県の年齢構成別人口について, ある年から数年間にわたって取り出すという場合もあれば, 全市区町村の特定の年の年齢構成別人口を対象としたい場合も考えられるので, 柔軟な想定をしておかねばならない。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 (Postgre SQL バージョン 7.2⁴⁾) を用意し, それにアクセスするインターフェイスを web とエクセルで利用可能な ODBC⁵⁾機能を用いて, 構築することとした。なお, 学内からの利用のみを想定し, サーバに与えられた ip アドレスはプライベートネットワーク内の static としている。以下, 2 章では地域データの構造と特徴について, 3 章では具体的なデータベース作成のための手順, 4 章では利用するための手順について述べる。そして最後に, 今後に残された問題点について指摘したい。

2 地域データの構造と特徴

われわれが開発した地域データは日経メディアマーケティング社から本学が購入した日経地域総合ファイルを基にして構築されている。これは, 面積,

2) KG-edens では, 全データを 300 ほどのシーケンシャルなデータファイルに分割し, インデックスファイルから各ファイルを読み出していた。

3) Fedora Core2 詳しくは <http://fedora.jp/>を参照

4) 詳しくは, <http://www.postgresql.org/>を参照

5) Open DataBase Connectivity

人口、世帯など基礎的な地域データを網羅しており、さらに都道府県から市区町村までの全国自治体の普通会計決算も収録されている。具体的には、地域総合（土地・人口・世帯、就業基本調査、県民経済計算、生産農業所得、事業所・企業統計、金融・所得、生活・文化、住宅・地価など）、国勢調査（国勢調査男女別5歳階級別人口、人口集中地区別人口・面積、家族類型別世帯数、従業上の地位別就業者数、職業別就業者数、産業別就業者数、昼間人口、流出先・流入先別15歳以上通勤者数および通学者数など）、人口推計（2000年国勢調査をベースとした2005年から2030年までの男女別5歳階級別人口の推計値）、商業統計（商業統計表）、工業統計（工業統計表）、地方財政（都道府県、市区町村の普通会計決算、決算収支の状況、歳入内訳、地方税の内訳、目的別歳出内訳、性質別歳出内訳、地方債現在高など）である。

つまり、地域総合ファイルには面積、世帯数、人口など2000項目以上（人口に関するもの728項目、商業・工業統計742項目、財政に関するもの158項目、地域の総合データ477項目）が、合計3400以上の都道府県市区町村に対して（ただし各行政主体に対して定義されていないものは除く）、期間は最も長いもので1970年から2001年まで年次データとして収録されている（予測値は除く）。なお、国勢調査項目などは5年に1度だけのデータ収録である。

中身としては、表1にあるように、7つのファイルからなる。これらはおのおのgzip⁶⁾にて圧縮された状態で1枚のCD-ROMに1つのファイルが収録されているので、システム全体としては、7枚のCD-ROMからなっている。また付属として都道府県市区町村のJISコード表（5桁）、項目名のコード表もある。

表2にあるようにデータファイルは固定長データで、第1フィールドには都道府県コード、第2フィールドには市区町村と第3フィールドにある項目コードごとにレコードが構成されており、第5フィールドにある収録開始期のデータが第10フィールドに記録され、以下順次収録終了期まで収録されている。これは表1にあるすべてのファイルに共通である。具体的には、表3の

6) ファイル圧縮形式の一つ。unixなどでよく用いられるオープン・ソース・ソフトウェア。

ような形でデータが収録されている。

これを読むための perl のプログラムは表 4 であり、これによってどのようなデータがどれだけ収録されているかを見ることができる。特に都道府県のみに対して収録されているデータなどはいちいち印刷物を調べずとも、先のプログラムを perl01.pl として、

```
perl perl01.pl | grep P0001
```

のようにすれば容易に P0001 の収録状況を調べることができる。

3 データベースの構築手順

データベース構築の手順を以下に示す。STEP A～E という 5 つのステップを経ると、都道府県市区町村コード、項目コード、年度を指定すれば当該のデータを excel などでも利用可能な形に変換できる。

< STEP A >

表 1 の各ファイルを 1 つのレコードが

(県コード 市区町村コード 年月 項目コード)

で区切り子⁷⁾が tab⁸⁾である形式のファイルに変換する。なおこのとき、変換後のファイル名は元のファイル名の最後に”a”を付加したものとする。また同時にファイルサイズを小さくするために、データ部分が空白のものについては出力しないという処理も行う。このための perl によるプログラムリストが表 4 であり、加工されたファイルが表 5 である。ここで、このリストは、5 行目で入力ファイル、6 行目で出力ファイルを設定した後で、1 行ずつ読み込み (9 行目)、11 行目から 43 行目で、それらのデータを適切に分割する。特に 23 行目から 42 行目はデータ自体が格納される。45 行目では、データが複数レコードにわたって収録されている際に用いる調整値を設定している。この値は 1 つのレコードに 20 個のデータが収録されていることによる。46 行目から 54 行目までが実際の出力部分で、47 行目で期間の設定、48 行目で出力するデータ

7) delimiter という

8) ¥x09

の仮設定を行い、実際に書き出すのはこのデータが15個のブランク以外からなるときだけであり、書き込むときには、進捗確認用に画面にも出力し（51行目）、52行目で実際のファイルへの書き込みを行っている。そして以上の処理を、ファイルの最後にたどり着くまで継続する（55行目）。なおこれらの処理を実行する際には、あらかじめハードディスクにテンポラリーファイルを作るためのかなりの余裕（おおむねもとのデータの2倍程度）が必要である。

< STEP B >

STEP A で作成したファイルを結合して1つのファイル torades01.data を作成する。このための shell script が表 6 である。

< STEP C >

次に、これをデータベースソフトに読み込ませると、とりあえず SQL コマンドが利用可能なサブシステムが完成する。そのためのスクリプトが表 7 である。これは1行目で今から作成するファイルを初期化し、2行目から8行目まででファイル構造を規定している。つまり第1フィールドが都道府県コード、第2フィールドが市区町村コード、第3フィールドが項目コード、第4フィールドが収録年、そして第5フィールドにデータが、すべて文字列(text)形式で格納される。そして9行目で先に作成したファイルを区切り子を tab として読み込み、最後に項目コードをキーにしてインデックスを作成するために10行目を追加した。

オリジナル ファイル名	STEP A 後の ファイル名	内 容
CENT00	CENT00a	地域総合
POPU10	POPU10a	人口
POPU20	POPU20a	人口動態
POPU30	POPU30a	推計人口
COMM00	COMM00a	商業統計
INDUS0	INDUS0a	工業統計
FINA00	FINA00a	地方財政

表 1: 地域データのファイル名と内容

同様に、表 8 の形式をしている各項目コードと項目名の対照表も、スクリプト表 9 によってデータベース化⁹⁾され、さらに表 10 の形式をもつ地方自治体コードと都道府県市区町村名の対照表もスクリプト表 11 によってデータベース¹⁰⁾に取り込まれる。

< STEP D >

これらを組み合わせて、実際に抽出したときに利用しやすいようにするためのステップが STEP D である。スクリプト表 12 にあるように、1 行目でファイルの初期化をした後、2 行目で次のようにレコードの追加を行っている

- 1) 市区町村コードの後に文字列の都道府県名+市区町村名を入れる
- 2) 項目コードの後に文字列の項目名を入れる

この処理は、ユーザが RDB¹¹⁾を直接操作できる環境のところでは、データベース間の構造がシンプルである本システムの場合、ユーザサイドで作業できないこともない。しかし excel では利用法がかならずしもわかりやすすくないので、1 つのデータベースにまとめることにした。しかしこのために、アクセス速度の低下やファイルサイズの肥大化というデメリットも生じた。なおこの行にある trim コマンドは読み込んだ文字列に空白などが含まれているときにそれを取り除くためのコマンドであり、"||" は文字列結合のコマンドである。これによって、都道府県コード+市区町村コードの比較が行われている。最後に 3,4 行目はアクセス速度向上のためのインデックス作成コマンドであり、順に項目名、収録年を対象としている。

< STEP E >

本システムの場合、一般ユーザには参照権限¹²⁾を与えることなのでスクリプト表 13 にあるコマンドを発行する必要がある。また、これとは別に、本サーバに対するアクセスをどのマシンから許すのかも検討しておく必要がある。イントラネット全体からの利用を許すのならば、pg_hba.conf において、`"local torades-j all trust"` といった設定が必要である¹³⁾。

9) torades_header

10) torades_jis2name

11) Relational DataBase

12) PostgreSQL では select 権限という

13) これは PostgreSQL そのものの管理者が行う

項目名	桁数
県コード	2
市区町村コード	3
項目コード	5
レコード番号	2
収録開始期	6
収録終了期	6
期種	2
小数部桁数	2
変換コード	2
データフラグ	20
データ 1	15
データ 2	15
⋮	⋮
データ 15	15

表 2: オリジナルデータの構造

4 利用の方法

まず、クライアントマシン¹⁴⁾の設定を行う。そのためには、管理者権限が必要である。OSのバージョンによって異なる部分もあるが、必要な事項は共通している。Windows XP Pro を例にとると以下のとおり。

1) ODBC ドライバのインストール

psqlodbc.dll の取得¹⁵⁾とレジストリへの登録
ダウンロードしたファイルを展開し、指示書にあるとおりファイルの複写とレジストリへの登録を行う。

14) ここでは、Windows XP Pro, Windows 2000 を想定している

15) 例えば、<http://www.interwiz.koganei.tokyo.jp/software/PsqlODBC/index.html> からダウンロードできる

```

01000_P000101197000200000CA00_L.....0.....0.....0.....+00000005683062
.....+0000000569232
1.....+000000056436
47.....+00000005679
439..... (改行)
01000_P000102197000200000CA00_L.....0.....0.....0.....+00000005575989
.....+0000000533820
6.....+000000051842
87..... (改行)
(中略)
01000_C000601197900200000FA00_S.....00000000000000000000+0000000046124
+00000000046745+00000000048667+00000000049112+00000000049773+0000000004957
7+00000000052281+00000000050994+00000000052787+00000000053985+000000000541
27+00000000054824+00000000058340+00000000060769+00000000063246+00000000065
825+00000000069203+00000000071668+00000000072997+00000000071629_ (改行)
01000_C000602197900200000FA00_S.....00.....+00000000074541
+00000000078133.....
.....
..... (改行)
(以下略)

```

表 3: オリジナルデータ (改行までが 1 行) 上段; 北海道全体の人口, 下段; 北海道全体の<住民票記載数>出生者数

2) ODBC ドライバの登録

コントロールパネルから管理ツール, データソースと順に開き, 先にレジストりに登録された PostgreSQL を ODBC ドライバに登録する。

3) 設定 (ip アドレス, データベース名)

ユーザー DSN の画面において, ユーザーデータでソース追加を行う。User Name は PostgreSQL 上で createuser によってユーザ登録しておく必要があり, またここで用いた ip アドレス 172.20.30.40 は架空のものである。

Database	toradesj
Server	172.20.30.40
User Name	torades


```

01: #全項目を
02: #県 \t 市区町村 \t 項目 \t 期 \t data
03: #にスライスする
04: #####
05: open (IN0,"<CENT00");
06: open (OUT,">CENT00a");
07: #####
08: do {
09: read (IN0,$xx,362);
10:
11: $a[1] =substr($xx, 0, 2);#県コード
12: $a[2] =substr($xx, 2, 3);#市区町村
13: $a[3] =substr($xx, 5, 1);#ブランク
14: $a[4] =substr($xx, 6, 5);#項目
code
15: $a[5] =substr($xx, 11, 2);#rec No.
16: $a[6] =substr($xx, 13, 6);#開始期
17: $a[7] =substr($xx, 19, 6);#終了期
18: $a[8] =substr($xx, 25, 2);#期種
19: $a[9] =substr($xx, 27, 2);#小数部桁
20: $a[10]=substr($xx, 29, 2);#変換
code
21: $a[11]=substr($xx, 31, 8);#予備
22: $a[12]=substr($xx, 39,20);#data_flg
23: $a[13]=substr($xx, 59,15);#データ 01
24: $a[14]=substr($xx, 74,15);#データ 02
25: $a[15]=substr($xx, 89,15);#データ 03
26: $a[16]=substr($xx,104,15);#データ 04
27: $a[17]=substr($xx,119,15);#データ 05
28: $a[18]=substr($xx,134,15);#データ 06
29: $a[19]=substr($xx,149,15);#データ 07
30: $a[20]=substr($xx,164,15);#データ 08
31: $a[21]=substr($xx,179,15);#データ 09
32: $a[22]=substr($xx,194,15);#データ 10
33: $a[23]=substr($xx,209,15);#データ 11
34: $a[24]=substr($xx,224,15);#データ 12
35: $a[25]=substr($xx,239,15);#データ 13
36: $a[26]=substr($xx,254,15);#データ 14
37: $a[27]=substr($xx,269,15);#データ 15
38: $a[28]=substr($xx,284,15);#データ 16
39: $a[29]=substr($xx,299,15);#データ 17
40: $a[30]=substr($xx,314,15);#データ 18
41: $a[31]=substr($xx,329,15);#データ 19
42: $a[32]=substr($xx,344,15);#データ 20
43: $a[33]=substr($xx,359, 1);#予備
44:
45: $ik=($a[5]-1)*20;
46: for ($k=0; $k<=19; $k++){
47: $period=$a[7]-100*($k+$ik);
48: $data=$a[$k+13];
49: if($data ne " ") {
50: $data=$data*1;
51: print "$a[1]$a[2],";
52: print OUT "$a[1] \t $a[2] \t
$a[4] \t $period \t $data\n";
53: }
54: }
55: } while (eof(IN0)!=1);
56: close (IN0);

```

表 4: STEP A;perl の変換プログラム

01 000 P0001 197000 5683062	01 000 C0006 197900 46124
01 000 P0001 197500 5692321	01 000 C0006 198000 46745
01 000 P0001 198000 5643647	01 000 C0006 198100 48667
01 000 P0001 198500 5679439	01 000 C0006 198200 49112
01 000 P0001 199000 5575989	01 000 C0006 198300 49773
01 000 P0001 199500 5338206	01 000 C0006 198400 49577
01 000 P0001 200000 5184287	01 000 C0006 198500 52281
(中略)	(以下中略)

表 5: STEP A の変換後のデータ

```
01:cat CENT00a > torades.data
02:cat POPU10a >> torades.data
03:cat POPU20a >> torades.data
04:cat POPU30a >> torades.data
05:cat COMM00a >> torades.data
06:cat INDUS0a >> torades.data
07:cat FINA00 >> torades.data
```

表 6: STEP B; 個別ファイルの torades.data への連結

```
01:drop table torades;
02:create table torades(
03:  pref text, --都道府県コード
04:  city text, --市町村コード
05:  item text, --項目コード
06:  year text, --収録年
07:  data float --データ
08:);
09:copy torades from 'torades.data' delimiters '\t';
10:create index torades_item on torades (item);
```

表 7: PostgreSQL へのデータ入カスクリプト

C0001	面積	C0211	全国銀行預金残高
C0003	世帯数	C0212	全国銀行貸出残高
C0004	住民票記載数	C0214	国内銀行数
C0005	<住民票記載数>転入者数	C0217	課税対象所得額
C0006	<住民票記載数>出生者数	C0239	水道・年間総配水量
C0007	<住民票記載数>その他	C0240	都市公園面積
C0008	住民票消除数	C0241	病院（施設数）
C0009	<住民票消除数>転出者数	C0242	病院（病床数）
C0010	<住民票消除数>死亡者数	C0243	一般診療所（施設数）
C0011	<住民票消除数>その他	C0244	一般診療所（病床数）
C0012	婚姻件数	C0245	歯科診療所
C0013	離婚件数	C0246	小学校（学校数）
C0014	<人口>男	C0247	中学校（学校数）
C0015	<人口>女	C0248	高等学校（学校数）
C0059	生産農業所得		（以下略）

表 8: 項目コード，項目名対応表

```

01:drop table torades_header;
02:create table torades_header(
03:  item_name text,
04:  name text
05:);
06:copy torades_header from 'torades.head'; delimiters '\t';
07:create index torades_header_item_name on torades_header (item_name);

```

表 9: 項目コード，項目名対応表のための sql

これによって、ローカルマシンから、データベースサーバへの設定がなされた。なお、ユーザは grant コマンドで許した範囲においては、パスワードなどの認証なしでデータベースを利用できる。

次に、クライアントからの利用方法について述べる。excel の場合には、ツール→データ→外部データの取り込み→新しいデータベースクエリで、抽出するデータベース（許可を受けたもの）にアクセスし、条件を設定するとワーク

01000	<北海道>	(中略)
01100	札幌市	47349 大里村
01101	中央区	47350 南風原町
01102	北区	47351 仲里村
01103	東区	47352 具志川村
01104	白石区	47353 渡嘉敷村
01105	豊平区	47354 座間味村
01106	南区	47355 粟国村
01107	西区	47356 渡名喜村
01108	厚別区	47357 南大東村
01109	手稲区	47358 北大東村
01110	清田区	47359 伊平屋村
01202	函館市	47360 伊是名村
01203	小樽市	47371 城辺町
01204	旭川市	47372 下地町
01205	室蘭市	47373 上野村
01206	釧路市	47374 伊良部町
01207	帯広市	47375 多良間村
01208	北見市	47381 竹富町
01209	夕張市	47382 与那国町
01210	岩見沢市	99999 <全国計>
01211	網走市	

表 10: 地方自治体 (JIS) コード, 名称対応表

```

01:drop table torades_jiscode2name;
02:create table torades_jiscode2name(
03:  code text, --項目コード
04:  name text --項目名
05:);
06:copy torades_jiscode2name from 'jis-code.data' delimiters '\t';
07:create index torades_jiscode2name_item
    on torades_jiscode2name (code);
08:create index torades_jiscode2name_name
    on torades_jiscode2name (name);

```

表 11: 地方自治体 (JIS) コード, 名称対応表作成のための sql

```

01: drop table torades_j;
02: SELECT pt.pref,pt.city,trim(js.name) as name2,pt.item_name,
        trim(ji.name), pt.year,pt.data  into torades_j
        from torades_jiscode2name as js,
        torades as pt,torades_header as ji
        where js.code=trim(pt.pref) || trim(pt.city)
        and ji.item_name=trim(pt.item_name);
03: CREATE INDEX torades_j_item on torades_j (item_name);
04: CREATE INDEX torades_j_year on torades_j (year);

```

表 12: データベース結合のための sql コマンド

```

01: GRANT SELECT ON torades_j TO public;

```

表 13: データアクセス許可のための sql コマンド

シート上に必要なデータを転送することができる。また access の場合にはファイル→外部データの取り込み→インポートでファイルの種類を ODBC データベースにする。ここでは、実際に 1) 西宮市の総人口の推移を求める場合、2) 1999 年における各都道府県の県民所得を取り出す場合の手順について述べる。

また、excel からはツール→データ→外部データの取り込み→新しいデータベースクエリという設定手順で、抽出するデータベース（許可されたもの）にアクセスし条件を設定すると、ワークシート上に必要なデータを転送することができる。以下ではその方法を述べる。

具体的には、西宮市の人口を取り出す手順を示す。まず、西宮市の都道府県市区町村コード、総人口の項目コードを調べ、それからデータの取り出しを行う。

- 1) サーバ上にある torades_jiscode2nam を選択して、西宮市のコード番号を調べる。そのために、name フィールドに“‘西宮’を含む”を設定し、対応する code(28204) を取り出す。なお、この上位 2 桁は都道府県コードを示す。
- 2) “男女の人口総数”のコードを調べるために torades_header を選択し、name フィールドに“‘人口’を含む”レコードを出力させその中から、目的のコード

である P00001 を選び出す。

3) torades_j において、pref¹⁶⁾=28,city¹⁷⁾=204、itemname¹⁸⁾=P0001 を設定した上で、出力する順序を year¹⁹⁾順とするように設定すると、図 1 のような結果が得られる。

さらに、全都道府県の 1999 年の県民所得を求める 2 段階の手順を示す。

1) 県民所得のコードを調べるために、torades_header の name フィールドに“県民所得”と入力し、C1254 が目的のコードであることを明らかにする。

2) 抽出年が 1999 年なので、torades_j において、year=199900²⁰⁾とし、さらに先の場合と同様に itemname=C1254 とする。そして出力順序を pref 順にすれば、都道府県順に並べられた図 2 のような結果が得られる。

もちろん、実際に本システムがインストールされているサーバに直接ログインしてデータベースを利用することも可能であるが、その場合には当該サーバ管理者の許可（そのマシンの ID、パスワード発行、さらには PostgreSQL 利用のための id 設定）が必要となる。学内の場合、samba²¹⁾などのファイル共有の環境が整っている範囲では、ファイル転送なども比較的容易に利用でき

	A	B	C	D	E	F	G
1	pref	city	name2	item_name	btrim	year	data
2	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	197000	377043
3	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	197500	400622
4	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	198000	410329
5	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	198500	421267
6	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	199000	426909
7	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	199500	390389
8	28	204	西宮市	P0001	[年齢別人口]総数(男女計)	200000	438105

図 1: エクセルへの読み込み (1)

16) 都道府県コード

17) 市区町村コード

18) 項目コード

19) 年月

20) 下 2 桁は月次用に用意されているが、期種が年次である本データベースにおいては、すべて 00 となる。

21) UNIX および UNIX 互換マシンを Windows NT/2000 互換のファイルサーバ/プリント・サーバにするオープン・ソース・ソフトウェア

	A	B	C	D	E	F	G
1	pref	city	name2	item_name	trim	year	data
2	01	000	<北海道>	C1254	[県民所得(分配)]県民所得	199900	15460454
3	02	000	<青森県>	C1254	[県民所得(分配)]県民所得	199900	3663451
4	03	000	<岩手県>	C1254	[県民所得(分配)]県民所得	199900	3765146
5	04	000	<宮城県>	C1254	[県民所得(分配)]県民所得	199900	6638649
6	05	000	<秋田県>	C1254	[県民所得(分配)]県民所得	199900	3080843
7	06	000	<山形県>	C1254	[県民所得(分配)]県民所得	199900	3313458
8	07	000	<福島県>	C1254	[県民所得(分配)]県民所得	199900	6000558
9	08	000	<茨城県>	C1254	[県民所得(分配)]県民所得	199900	9080320
10	09	000	<栃木県>	C1254	[県民所得(分配)]県民所得	199900	6382520
11	10	000	<群馬県>	C1254	[県民所得(分配)]県民所得	199900	6204802
12	11	000	<埼玉県>	C1254	[県民所得(分配)]県民所得	199900	22687930
13	12	000	<千葉県>	C1254	[県民所得(分配)]県民所得	199900	19019678
14	13	000	<東京都>	C1254	[県民所得(分配)]県民所得	199900	49591149
15	14	000	<神奈川県>	C1254	[県民所得(分配)]県民所得	199900	27546526
16	15	000	<新潟県>	C1254	[県民所得(分配)]県民所得	199900	7358354
17	16	000	<富山県>	C1254	[県民所得(分配)]県民所得	199900	3299161
18	17	000	<石川県>	C1254	[県民所得(分配)]県民所得	199900	3549437
19	18	000	<福井県>	C1254	[県民所得(分配)]県民所得	199900	2352737
20	19	000	<山梨県>	C1254	[県民所得(分配)]県民所得	199900	2544852
21	20	000	<長野県>	C1254	[県民所得(分配)]県民所得	199900	6644904
22	21	000	<岐阜県>	C1254	[県民所得(分配)]県民所得	199900	6129794
23	22	000	<静岡県>	C1254	[県民所得(分配)]県民所得	199900	11708107
24	23	000	<愛知県>	C1254	[県民所得(分配)]県民所得	199900	24878129
25	24	000	<三重県>	C1254	[県民所得(分配)]県民所得	199900	5268023
26	25	000	<滋賀県>	C1254	[県民所得(分配)]県民所得	199900	4301011
27	26	000	<京都府>	C1254	[県民所得(分配)]県民所得	199900	7771272
28	27	000	<大阪府>	C1254	[県民所得(分配)]県民所得	199900	29149131
29	28	000	<兵庫県>	C1254	[県民所得(分配)]県民所得	199900	16062350
30	29	000	<奈良県>	C1254	[県民所得(分配)]県民所得	199900	4036870
31	30	000	<和歌山県>	C1254	[県民所得(分配)]県民所得	199900	2676945
32	31	000	<鳥取県>	C1254	[県民所得(分配)]県民所得	199900	1595107
33	32	000	<島根県>	C1254	[県民所得(分配)]県民所得	199900	1881928
34	33	000	<岡山県>	C1254	[県民所得(分配)]県民所得	199900	5372109
35	34	000	<広島県>	C1254	[県民所得(分配)]県民所得	199900	8533302
36	35	000	<山口県>	C1254	[県民所得(分配)]県民所得	199900	4356137
37	36	000	<徳島県>	C1254	[県民所得(分配)]県民所得	199900	2259849
38	37	000	<香川県>	C1254	[県民所得(分配)]県民所得	199900	2886440
39	38	000	<愛媛県>	C1254	[県民所得(分配)]県民所得	199900	3640945
40	39	000	<高知県>	C1254	[県民所得(分配)]県民所得	199900	1946047
41	40	000	<福岡県>	C1254	[県民所得(分配)]県民所得	199900	13750244
42	41	000	<佐賀県>	C1254	[県民所得(分配)]県民所得	199900	2357324
43	42	000	<長崎県>	C1254	[県民所得(分配)]県民所得	199900	3612192
44	43	000	<熊本県>	C1254	[県民所得(分配)]県民所得	199900	4723542
45	44	000	<大分県>	C1254	[県民所得(分配)]県民所得	199900	3272591
46	45	000	<宮崎県>	C1254	[県民所得(分配)]県民所得	199900	2742722
47	46	000	<鹿児島県>	C1254	[県民所得(分配)]県民所得	199900	4172546
48	47	000	<沖縄県>	C1254	[県民所得(分配)]県民所得	199900	2846750
49	99	999	<全国計>	C1254	[県民所得(分配)]県民所得	199900	390126336

図 2: エクセルへの読み込み (2)

る。ユーザが torades サーバにログインする権限を持っているときには、そして北海道の都道府市区町村コードが"01000", 人口のコードが"P0001"であることがあらかじめ判明しているとき, 次のような SQL コマンドによって北海道の各年の人口を取り出すことができる。もちろん、これだけでは結果が画面に表示されるだけなので、その結果をファイルに取り込むためには別のコマンド²²⁾を発行する必要がある。

```
SELECT * from torades_j where pref="01" and city="000"
and item_name="P0001" order by year;
```

5 今後の課題

2001 年度より活発に行われ始めた市町村合併²³⁾によって、本システムも大きな影響を受けた。名称変更によるコード番号のずれと過去データとの整合性など、最低限の対応しかできていないのが実態である。もちろん市区町村数の減少によって、データの取り出しの手続きは変更されないが、継続性をどのように考えるかによって、1) 年度ごとに別のデータベースを作成する、2) 市区町村コードごとに別のデータを作成し、合併以前と以後を試行的に接続させたもの作成するといった方策を採ることができる。

技術的な問題に加えて、現実的な利用のしやすさも視野に入れたシステム構築をめざさねばならない。情報機器、特にストレージ環境の進歩を見ると本システムそのものをローカルに持った上で作業するというのも十分現実的になってきている。そのときには、本システムのようにサーバにデータを保存するスタイルがいいのか、ローカルのマシンにデータを格納するほうがいいのかも検討しなければならない。もちろんその際には、データおよびシステムの著作権管理、利用範囲などを明確にすべきであることは言うまでもない。

22) /w など

23) <http://www.kokudo.or.jp/new/cities/>を参照

<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 年 11 月。

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 年 3 月。

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 年 08 月。

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 年 09 月。

< PostgreSQL に関するもの >

石井 達夫編, PCUNIX ユーザのための Postgre SQL 完全攻略ガイド (改訂第 4 版), 技術評論社, 2004 年 06 月。