

金融関連会社に関する日経 NEEDS 財務データに もとづくデータベースの構築

地 道 正 行

I はじめに

本稿で扱うテーマは、拙稿（地道（2010-a, b））で扱った「一般事業会社」の学内向けの財務データ抽出システム KGUSBADES¹⁾を「金融関連会社」を含めたものに拡張することにある²⁾。具体的には、2010年版「日経 NEEDS 財務データ」MT³⁾版で提供されている「銀行」、「証券（会社）」、「保険（会社）」の財務データを利用したデータベースと抽出システムを構築する。

本稿の構成は以下のようなものである。まず、II節では「銀行」を例としたデータベースの構築に関して述べる。次に、III節において、Webサーバ Apache⁴⁾とMySQL⁵⁾をPHP⁶⁾を用いて連携させることによってWebブラウザを使ってデータ抽出を可能とするシステムについて述べるとともに、IV節において、このデータ抽出サービスを実際に利用する例を与える。特に、データ解析環境 R⁷⁾からデータベースにODBC⁸⁾を使ってアクセスし、データ抽

-
- 1) Kwansai Gakuin University, School of Business Administration, Data Extraction System の略。
 - 2) なお、詳細は割愛するけれども、今回の構築に際して地道（2010-a, b）で扱われた「一般事業会社」に関するデータもまた2010年版のものにバージョンアップしていることに注意しよう。
 - 3) Magnetic Tape の略。
 - 4) <http://www.apache.org/>
 - 5) <http://www.mysql.com/>
 - 6) <http://www.php.net/>
 - 7) <http://www.r-project.org/>

出を行うことに言及する。最後に、V節においてまとめと今後の展望について述べる。なお、付録には日経 NEEDS 財務データ MT 版の解説とそれにもとづいて構築されたデータベースの構造や構築に利用される各種のスク립トのソースコードのリスト等が与えられる。

II データベースの構築

今回構築するデータベースは、「銀行」、「証券（会社）」、「保険（会社）」に関する3種類のデータと、それらの「連結本決算」と「単独本決算」に関するもの、さらに、開発プラットフォームとして、Mac OS X Server 10.5, Ubuntu⁹⁾ 11.04, Windows 7 の3種類のものが存在するけれども、簡単のため「銀行」に関する連結本決算にもとづくデータベースの構築をプラットフォームに共通した手順のみに限定して述べることにする。なお、LAMP¹⁰⁾ 環境が設定済みであることを前提としていることに注意しよう。(LAMP 環境の構築に関しては、たとえば、地道 (2010-a, b) を参照のこと。)

まず、データベースの設計に関して以下のことに注意する。

「銀行」に関する連結本決算のデータベースの設計項目

(設計項目1) 連結本決算データ(日経によるコード: HBFSCA050) に対するデータベース(hbfscA050)の構造として、レコード種別(FA11, ..., FG11)に対応するテーブル(fa11, ..., fg11)を持つものを作成する。

(設計項目2) 社名から日経会社コードなどを検索可能にするために社名リストのテーブル(firmList)を作成する。

ここで日経 NEEDS のデータに関するコードの詳細等については付録Aを、またデータベースに関するテーブルのコードなどの情報を含む詳細な構造については付録Bを参照されたい。

8) Open DataBase Connectivity の略。

9) <http://www.ubuntu.com/>

10) Linux, Apache, MySQL, PHP の略。正確には OS が Windows であれば WAMP, MacOS X のときは、MAMP となるけれども、統一的な表記として LAMP を採用することに注意しよう。

各手順で利用されるスクリプトファイルは表1のようなものである。なお、これらのファイルのソースコードの一部については、付録Dで与えられているので参照されたい。

表1：データベース構築のためのスクリプトファイル

ファイル名	説明
script.sh	データファイルの前処理を行うシェルスクリプトからなるファイル
separate.pl	ファイルにおける適切な箇所にフィールドを区切るセパレータを挿入するための Perl スクリプトファイル
loadtoMySQL.sql	データベースとテーブルを作成し、script.sh によって作成されたテーブル単位のデータファイルを MySQL にロードするための SQL スクリプトファイル
loadfirmlisttoMySQL.sql	社名リストファイル firmlistbank.txt を MySQL へロードするための SQL スクリプトファイル

以下に、「銀行」を例としてデータベース構築の手順を簡潔に与える¹¹⁾。

(手順1) ファイルの前処理

シェルスクリプト・ファイル script.sh と圧縮された連結本決算のデータファイル

HBFSCA050.2000.gz

HBFSCA050.2010.gz

を適当なディレクトリに保存後、ターミナルを起動し、そのディレクトリに移動した後、以下のように script.sh を実行する。(なお、\$ はシェルプロンプトである。)

```
$ /bin/sh script.sh
```

このスクリプトは、以下の操作を行う。

(a) unzip コマンドでデータファイルを展開した後、リダイレク

11) ここで与える手順は地道 (2010-a, b) に関するものとはほぼ同様であることに注意しよう。

ション (>, >>) を使ってこれらを1つのファイル `renketsu.all` にまとめる。

- (b) `grep`, `sed`, `Perl` 等のコマンドやインタプリタを使って、このファイルから `FA11`,...,`FG11` という文字列を含む行を個別に抜き出す。
- (c) `Perl` スクリプトファイル `separate.pl` を読み込んで、各ファイルのフィールド間に `tab` (`'\t'`) をセパレータとして挿入したファイル `FA11tab.data`,...,`FG11tab.data` を作成する。
(図1も参照。)

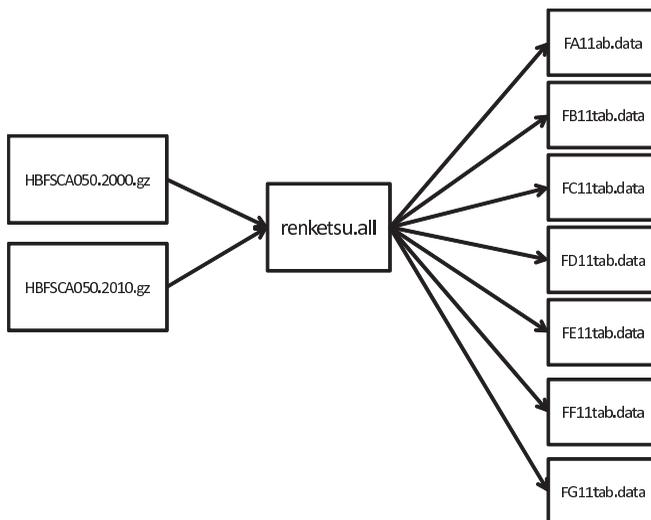


図1：データファイルの前処理

なお、`Mac OS X Server` と `Ubuntu` においてファイル `FA11tab.data`,...,`FG11tab.data` を作成する最終段階で文字コードを `UTF-8` に変換しておく必要があることに注意しよう。なお、詳細は地道 (2010-a, b) を参照されたい。

(手順2) ファイルの移動

(手順1)で作成したファイル FA11tab.data, ..., FG11tab.data を MySQL のルートディレクトリに移動 (またはコピー) する。なお、OS 毎の MySQL のルートディレクトリはデフォルトで以下のようなものである。

OS	MySQL のルートディレクトリの場所
Mac OS X Server	/var/mysql
Ubuntu	/var/lib/mysql
Windows	C:\VertrigoServ\MySQL\data

(手順3) データベースとテーブルの作成

ターミナルで loadtoMySQL.sql が存在するディレクトリに移動し、mysql コマンドを使って、MySQL モニタを root 権限で起動後、SOURCE コマンドを以下のように実行する。

```
mysql> SOURCE loadtoMySQL.sql
```

ここで、mysql> は MySQL のプロンプトである。loadtoMySQL.sql は以下の操作を行うスクリプトである。

- (a) CREATE コマンドでデータベース hbfsc050 とテーブル fa11, ..., fg11 を作成する。
- (b) LOAD コマンドを使って、(手順1)で作成したデータファイル FA11tab.data, ..., FG11tab.data をテーブル fa11, ..., fg11 に読み込む。(図2も参照。)

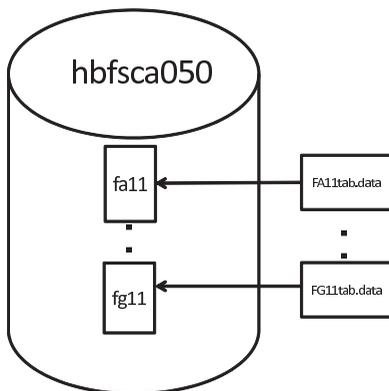


図2：データファイルのロード

(手順4) 社名リストのテーブルの作成

地道（2010-a, b）で行ったように社名リストのテーブル作成は Windows 上で行い，作成されたファイルを個々のプラットフォームのデータベースに読み込む手順で行う．なお，以下の手順において R による処理については地道（2010-a, b）で与えられた方法と全く同じであるため割愛する．

(L1) 会社情報のファイルへの出力

一旦作成したデータベース hbfsca050 のテーブル FA11 から，銀行の日経会社コード（a04），株価コード（a05），日経業種コード（a22），社名（銀行名）に関する情報（b071, ..., b080）を抽出しファイルへ書き出す．実際の作業は，MySQL モニタで以下のコマンドを実行する．（紙面の都合上，スクリプトの途中で改行していることに注意．）

```
mysql> USE hbfscsca050
mysql> SELECT a04,a05,a22,b071,b072,b073,b074,b075,
             b076,b077,b078,b079,b080 INTO OUTFILE
             'temp.txt'
             FIELDS TERMINATED BY '\t' FROM FA11;
```

(L2) R によるファイル処理

地道（2010-a, b）で与えられた方法を使って，R を用いた処理を行い，社名リストのファイル `firmlistbank.txt` を生成する。

(L3) 社名リストファイルのデータベースへの読み込み

ファイル `firmlistbank.txt` を MySQL のフォルダへコピー後，MySQL モニタから `SOURCE` コマンドを用いてスクリプトファイル `loadfirmlisttoMySQL.sql` を実行する。

```
mysql> SOURCE loadfirmlisttoMySQL.sql
```

(図 3 も参照.)

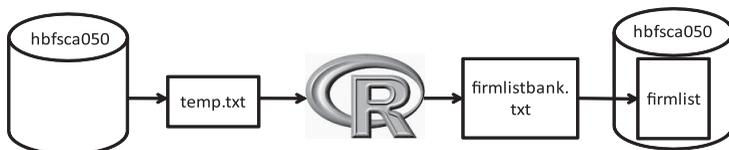


図 3：社名リストファイルのロード

(手順 5) データ抽出専用ユーザ `needs` の登録

セキュリティの観点から，多くのユーザがデータベースを利用することを考え，データ抽出専用のユーザ `needs` を登録する．MySQL モニタを起動し，`CREATE`，`GRANT` コマンドを用いて以下のように行う。

```
mysql> CREATE USER needs IDENTIFIED BY '*****';
mysql> GRANT SELECT ON hbfzca050.* TO 'needs'@'localhost';
```

ここで、***** は、ユーザ needs のパスワードを与える。

Ⅲ データ抽出システム

Ⅱ節で構築されたデータベースから Web 経由でデータを抽出するためのシステムにおいてサービス手順とその仕様は地道（2010-a, b）で与えられたものと同様であるので概要を表す図 4 と各仕様で利用される主要なファイルの説明を表 2 に与えるにとどめる。

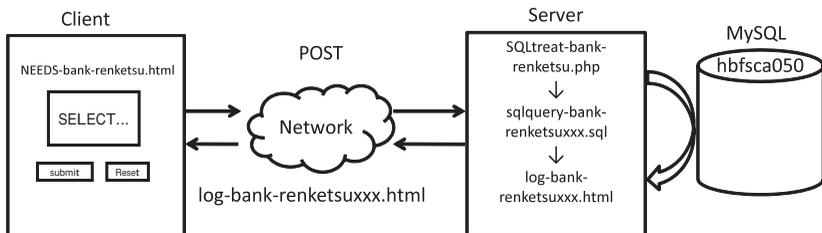


図 4：データ抽出システムの概要

表 2：データ抽出システムのためのスクリプト

ファイル名	説明
NEEDS-bank-renketsu.html	SQL 問合せのスクリプトをサーバへ送信するための HTML ファイル FORM の機能を使って PHP スクリプト SQL-bank-renketsu.php へ SQL 問合せを送信 (POST) する.
SQLtreat-bank-renketsu.php	以下の役割を果たす PHP スクリプトファイル 1. NEEDS-bank-renketsu.html によって送信された SQL 問合せの受け取り 2. SQL 問合せの内容をシェルのプロセス ID 付きで SQL スクリプトファイル sqlquery-bank-renketsuxxx.sql に出力 3. ファイル sqlquery-bank-renketsuxxx.sql の内容を引数として mysql を実行することによって抽出結果を log-bank-renketsuxxx.html に出力 ここで, xxx にはシェルのプロセス ID が与えられる.
sqlquery-bank-renketsuxxx.sql	SQLtreat-bank-renketsu.php によって出力される SQL スクリプトファイル.
log-bank-renketsuxxx.html	SQLtreat-bank-renketsu.php によって出力される HTML ファイル.

IV データベースの利用

この節では、データベースを Web 経由で利用する場合と、R から ODBC 経由で利用する場合について述べる。なお、両方ともデータの抽出には SQL 問合せ (SQL query) を記述することによって利用する方式をとることに注意しよう。また、SQL 問合せの例を付録 C に与えているので参照されたい。

IV.1 Web による利用

サーバへのアクセスは、学内から Internet Explorer (以下 IE と略.) などの Web ブラウザで適切な URL にアクセスすればよい¹²⁾。図 5 は、IE を使ってデータ抽出システムのトップページにアクセスしたときのイメージであ

12) 現時点では以下の URL が利用可能である:

<http://thorin.kwansei.ac.jp/KGUSBADES/> (Ubuntu)

<http://aule.kwansei.ac.jp/KGUSBADES/> (Mac OS X Server)

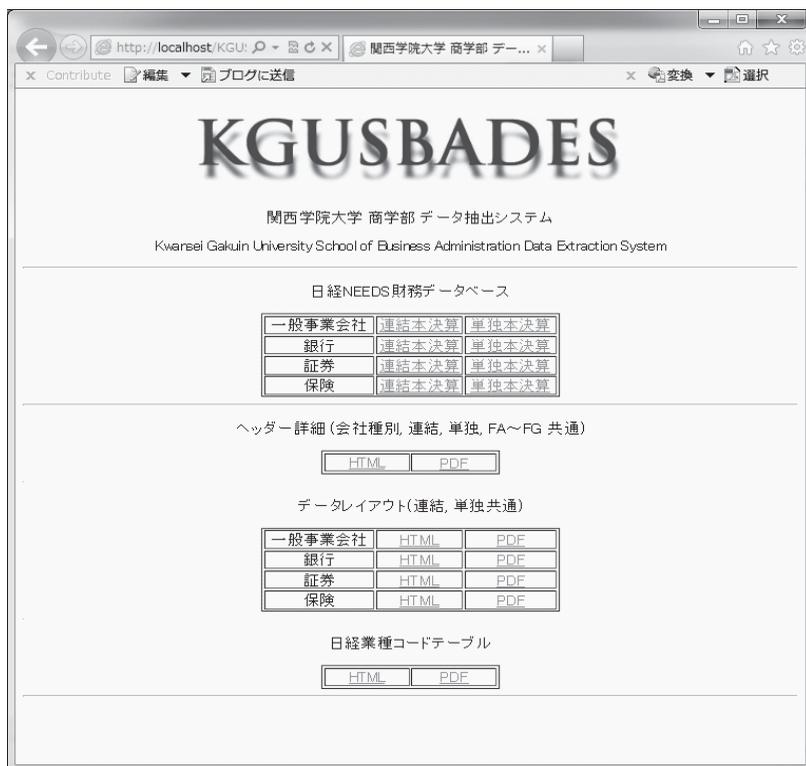


図5：IEでデータ抽出システムKGUSBADESにアクセスしたときのイメージ

る¹³⁾。

現時点で行われているサービスとしては日経NEEDS財務データにもとづく以下のようなデータ抽出であり、トップページにリンクが用意されている。

- 一般事業会社（連結本決算，単独本決算）
- 銀行（連結本決算，単独本決算）
- 証券（会社）（連結本決算，単独本決算）

13) 日々開発が進められているため、意匠は適宜変更される可能性があることに注意しよう。

- 保険（会社）（連結本決算，単独本決算）

また，データベースからデータを抽出する際に必要となる情報である各種のテーブルのファイルが HTML, PDF 形式でそれぞれ以下のように用意されているので，適宜参照されたい。

- ヘッダー詳細（連結，単独，FA～FG 共通）
- データレイアウト（連結，単独共通）
- 日経業種コードテーブル

なお，付録Bにデータベースの構造と各種のテーブルと同等の表を与えているのであわせて参照されたい。

データ抽出システム KGUSBADES のトップページから，たとえば「銀行」という行の「連結本決算」というリンクをクリックすることによって，銀行に関する連結本決算のデータを抽出するためのページにアクセスすることができる。図 6 は，IE を使ってページにアクセスしたときのイメージとその説明である。

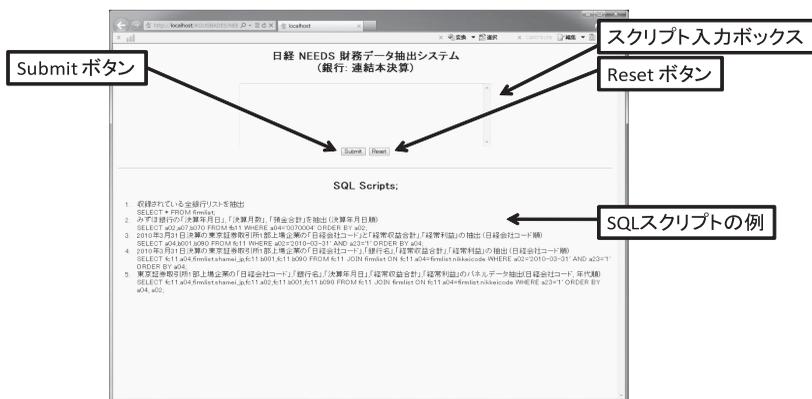


図 6：IE で銀行（連結本決算）に関するデータ抽出システムのページにアクセスしたときのイメージとその説明

実際にデータ抽出を行うには，「スクリプト入力ボックス」に SQL 問合せの スクリプトを入力し，「Submit」ボタンをクリックすることによって結果が画

面に表示される。

ここでは、収録されている全銀行の日経会社コード、社名等のデータを全て抽出することを例として、データ抽出システムの例を以下に与える。IE等の Web ブラウザを使って銀行の連結本決算データを抽出するシステムのトップページにアクセスした後、SQL スクリプトの例1 をスクリプト入力ボックスへコピー&ペーストし、`Submit` ボタンをクリックする。図7も参照せよ。

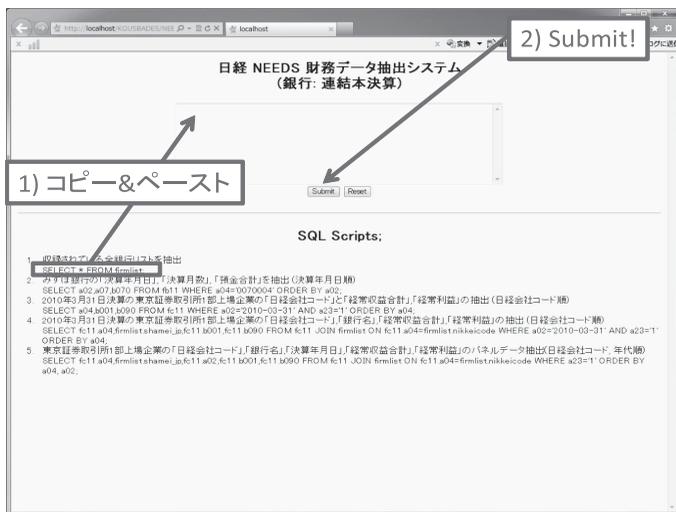


図7：データ抽出の例：収録されている全銀行の日経会社コード、社名等の抽出

以上のような手順によって、ブラウザ上に図8のような結果が表示される。

nikkeicode	stockcode	nikkeiyousuzocode	shamei_en	shamei_jp	shamei_jtk
000449	9999	252511	SHOKO CHAKIN BANK	商工組合中央銀行	シヨウコウカミアイヂユウオウキンコ
0081110	9999	247463	YRAYAKA HOLDINGS	豊ちやかホールディングス	チヨウヤカホールディングス
0081597	8400	247463	YACHYO BANK	八千代銀行	ヤチヨキンコウ
0070001	9999	247461	INDUSTRIAL BANK OF JAPAN	日本興業銀行	ニッポンコウギヨウキンコウ
0070002	8303	247462	SHINSEI BANK	新生銀行	シンセイキンコウ
0070003	8304	247462	ACCORA BANK	あおぞら銀行	アオゾラキンコウ
0070004	9999	247462	MZLHO BANK	みずほ銀行	ミズホキンコウ
0070005	9999	247462	SAKURA BANK	さくら銀行	サクラキンコウ
0070006	9999	247462	MZLHO CORPORATE BANK	みずほコーポレート銀行	ミズホコーポレートキンコウ
0070007	9999	247462	BANK OF TOKYO-MITSUBISHI UFJ	三菱東京UFJ銀行	ミツビシトウキョウエーエフジエイキンコウ
0070008	9999	247462	ASAHI BANK	あさひ銀行	アサヒキンコウ
0070009	9999	247462	UFJ BANK	UFJ銀行	ユーエフジエイキンコウ
0070010	9999	247462	SUMITOMO MITSUI BANKING	三井住友銀行	ミツイシミモキンコウ
0070011	9999	247462	RESORA BANK	りそな銀行	リソナキンコウ
0070012	9999	247462	TOKAI BANK	東海銀行	トウカイキンコウ
0070017	9999	247463	HOKKAIDO BANK	北海道銀行	ホツカイドウキンコウ
0070018	8342	247463	ACMORI BANK	青森銀行	アオモリキンコウ
0070019	8350	247463	MICHINOKU BANK	みちのく銀行	ミチノクキンコウ
0070020	8343	247463	AKITA BANK	秋田銀行	アキタキンコウ
0070021	9999	247463	HOKUTO BANK	北越銀行	ホクキョウキンコウ
0070022	9999	247463	SHINRI BANK	信利銀行	シヨウリエイキンコウ
0070023	8344	247463	YAMAGATA BANK	山形銀行	ヤマガタキンコウ
0070024	8345	247463	BANK OF IWATE	岩手銀行	イワテキンコウ
0070025	8349	247463	TCHOKU BANK	東北銀行	トウホウキンコウ
0070026	8341	247463	77 BANK	七十七銀行	シチジユウシキンコウ
0070027	8346	247463	TOHO BANK	東邦銀行	トウホウキンコウ
0070028	8354	247463	GUNMA BANK	群馬銀行	グンマキンコウ
0070029	9999	247463	ASHIKAGA BANK	足利銀行	アシカガキンコウ
0070030	8333	247463	JOYO BANK	常陽銀行	ジョウヨウキンコウ
0070031	8336	247463	TSURUBA BANK	筑波銀行	ツクバキンコウ
0070032	8336	247463	MUSASHINO BANK	武蔵野銀行	ムサシノキンコウ
0070033	8294	247463	CHIBA BANK	千葉銀行	チバキンコウ

図 8：データ抽出の結果：収録されている全行の日経会社コード、社名等の抽出結果

次に、2010年3月31日決算の東京証券取引所1部上場銀行の「決算年月日」と「経常収益合計」、「経常利益」を日経会社コード順に抽出し、MS Excelを用いて散布図の描画を行う例を以下に与える。なお、SQL問合せの詳細については付録Cを参照されたい。

まず、先の例と同様に、SQL 問合せ（**SQL Scripts:** の3番目の例）を **スクリプト入力ボックス** に入力し、**Submit** ボタンをクリックする。すると、抽出結果が表示されるので、**[編集]** メニューから **[すべてを選択]** を選び、さらに **[コピー]** を選ぶ。



図9：2010年3月31日決算の東京証券取引所1部上場銀行の「決算年月日」,
「経常収益合計」,「経常利益」の抽出とコピー

次に、MS Excel を起動後、適切な箇所にペーストし、列名を適切に変更後、「経常収益合計」,「経常利益」の列を選択し散布図を描く。

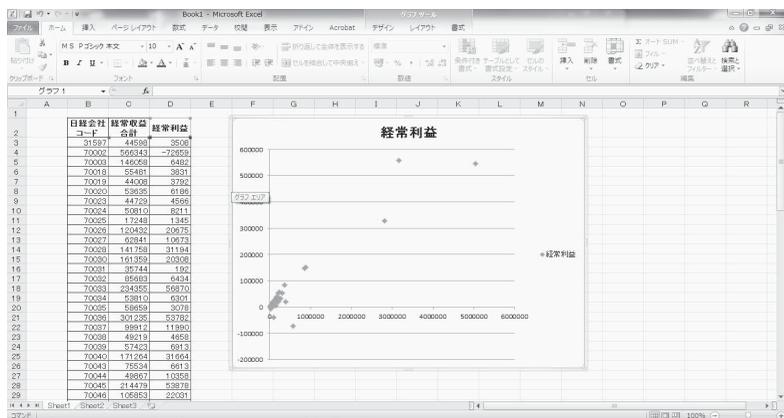


図10：2010年3月31日決算の東京証券取引所1部上場銀行の「経常収益合計」と
「経常利益」の散布図（列名を適切に修正していることに注意）

IV.2 R による利用

ここでは、データベースとの連携を行う仕組みであるオープン・データベース・コネクティビティ (Open Database Connectivity; ODBC) を R に実装したパッケージ RODBC を利用して銀行に関する連結本決算のデータベース

hbfsca050 からデータを抽出する例を与える。なお、ODBC と RODBC パッケージの説明に関しては、たとえば、スペクター (2008)、地道 (2010-a, b) を参照されたい。

データの抽出に関しては、以下のような入力を行ってパッケージの読み込みと接続の確立を行っておく必要があることに注意しよう。具体的には、R を起動し、[R Console] において library 関数を使って RODBC パッケージを読み込み、odbcConnect 関数を利用してデータベース hifaca052 と接続を確立する。

```
> library(RODBC)
> con.bank.renketsu<-odbcConnect('MyNEEDS-bank-renketsu')
```

ここで、> は R のプロンプトであり、MyNEEDS-bank-renketsu は ODBC ドライバのインストール時に登録した DSN¹⁴⁾ の名称である。なお、各種のデータベースに対応した DSN は以下の表のように与えられる。

表 3：データベースコードと DSN 名の対応

業種	DBコード (連結)	DSN	DBコード (単独)	DSN
一般事業会社	hifsc052	MyNEEDS-general-renketsu	hifsua052	MyNEEDS-general-tandoku
銀行	hbfsca050	MyNEEDS-bank-renketsu	hbfsua050	MyNEEDS-bank-tandoku
証券会社	hsfsc050	MyNEEDS-stock-renketsu	hsfsua050	MyNEEDS-stock-tandoku
保険会社	hnfsca050	MyNEEDS-insurance-renketsu	hnfsua050	MyNEEDS-insurance-tandoku

この入力によって、con.bank.renketsu オブジェクトに R と銀行に関する連結本決算のデータベース hbfsca050 との間に確立した接続に関する情報が格納される。この後の操作は、オブジェクト con.bank.renketsu の情報をもとに R の関数 sqlQuery を利用し、SQL 問合せをデータベースに送信することによってデータ抽出が行われることに注意しよう。

14) Data Source Name の略。ODBC において、データ (ベース) へのアクセスに必要な情報の照会に使用される論理名のこと。

みずほ銀行の預金合計の抽出（年月日順抽出）

みずほ銀行の「預金合計」を収録されている年度すべてに関して抽出することを例として与える。（みずほ銀行の日経会社コードが0070004であることに注意しよう。なお、誌面の都合上、行の途中で折り返している。）

```
> x<-sqlQuery(con.bank.renketsu,
'SELECT a02, a07, b070 FROM fb11 WHERE a04="0070004" ORDER BY a02')
```

ここでは、SQL 問合せをデータベースに送信し、結果をオブジェクト `x` に代入している。なお、`x` は「決算年月日」、「決算期間」、「預金合計」を列に持つ以下のようなオブジェクトである。

```
> x
      a02 a07      b070
1 2000-03-31 12 29129050
2 2001-03-31 12 30193291
3 2002-03-31 12 32693314
4 2003-03-31 12 48974345
5 2004-03-31 12 50407758
6 2005-03-31 12 50707935
7 2006-03-31 12 52304807
8 2007-03-31 12 53054306
9 2008-03-31 12 54435944
10 2009-03-31 12 55312169
11 2010-03-31 12 55718435
```

このように抽出されたデータは以下のように入力することによって時系列プロットを描くことができる。

```
> plot(as.Date(x[,1]), x[,3], type="l", xlab="年月日", ylab="預金合計")
> title("みずほ銀行の預金合計")
```

なお、このような入力によって [R Graphics] ウィンドウに図11のようなプロットが描画される。

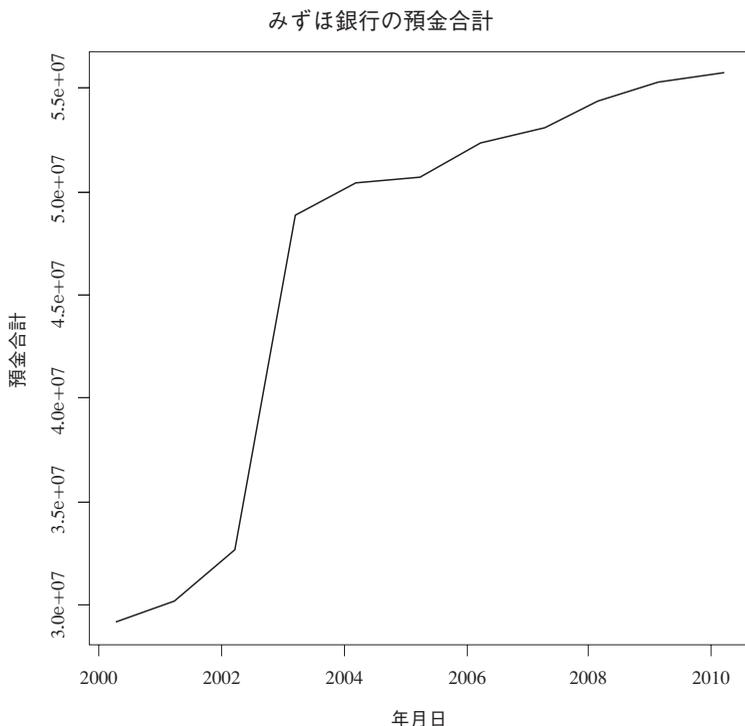


図11：みずほ銀行の預金合計の時系列プロット

東証1部上場銀行の経常収益合計と経常利益のパネルデータ抽出

東証1部上場銀行の「日経会社コード」, 「銀行名」, 「決算年月日」, 「経常収益合計」, 「経常利益」のパネルデータを「日経会社コード」と「決算年月日」順に抽出し、オブジェクト `y` に代入する例を以下に与える。

```
> y<-sqlQuery(con.bank.renketsu,
'SELECT fc11.a04,firmlist.shamei_jp,fc11.a02,fc11.b001,fc11.b090 FROM fc11
JOIN firmlist ON a04=firmlist.nikkeicode WHERE a23="1" ORDER BY a04,a02')
```

ここで、オブジェクト `y` は、「日経会社コード」(`fc11.a04`)と「銀行名」の漢字略称(`firmlist.shamei_jp`), 「決算年月日」(`fc11.a02`), 「経常収益合計」(`fc11.b001`), 「経常利益」(`fc11.b090`)を列としてもつオブ

ジェクトであり、時間的な推移をもつ特定の個体群のパネルデータ (panel data) であることに注意しよう。

オブジェクト `y` は、850行、5列のデータフレーム・オブジェクトであり、列名を関数 `colnames` を使って与え、関数 `head` を使って先頭の6行を表示すると以下のようなになる。

```
> colnames(y) <- c("日経会社コード", "銀行名", "決算年月日", "経常収益合計",
"経常利益")
> head(y)
  日経会社コード   銀行名   決算年月日   経常収益合計   経常利益
1      31597 八千代銀行 2000-03-31      54217      3555
2      31597 八千代銀行 2001-03-31      53744     -8379
3      31597 八千代銀行 2002-03-31      50913     -1095
4      31597 八千代銀行 2003-03-31      47525     -7103
5      31597 八千代銀行 2004-03-31      46377      4056
6      31597 八千代銀行 2005-03-31      47734      5418
```

地道 (2010-b) にならって、このオブジェクトに対して、東証1部上場の銀行に関する財務指標の時間的な推移を「母集団」と「個体」の観点から考察するために以下のようなプロットを行う。

- (1) 「経常収益合計」と「経常利益」に関する散布図を単年度毎にプロット
 - (2) 収録されている個々の銀行に対する「経常利益」の時系列プロット
- これらのプロットを行うためには、アドオン・パッケージ `lattice`¹⁵⁾ を利用することが効果的である。パッケージのロードは以下のように行う。

```
> library(lattice)
```

このパッケージには高機能の散布図を描くための関数 `xypplot` が用意されている。

15) 多変量データに関する様々なグラフを作成するためのツールを備えたパッケージ、ショーカー (2009) 参照。

単年度毎の散布図のプロット：単年度毎の「経常収益合計」に対する「経常利益」の散布図を描くためには以下のように入力する。

```
> xyplot(経常利益~経常収益合計 | format(y$決算年月日,"%Y"),data=y,
  pch=20,type=c("p","g"))
```

ここでは、format 関数によってデータフレーム y の属性の一つである「決算年月日」を「年」("%Y")に変換したものを条件として与える (|) ことによって各年毎の「経常収益合計」に対する「経常利益」を対応させる（経常利益~経常収益合計）散布図を描くことを行っている。その際、プロットに利用される点の種類（20番が“●”に対応）を pch=20 で与え、プロットのタイプ (type) として点 ("p") と基準グリッド ("g") を与える指定を行っていることに注意しよう。

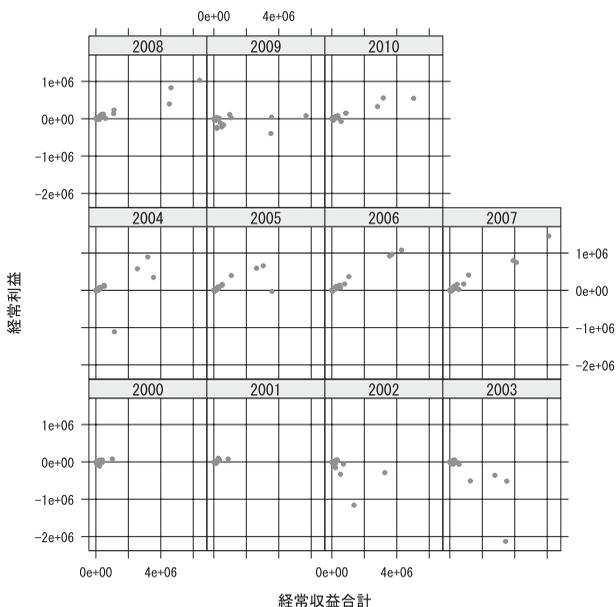


図12：2000年～2010年における東証1部上場銀行の単年度毎の「経常収益合計」と「経常利益」に関する散布図(時系列順プロット)

トリップ¹⁶⁾に表示されるテキストの大きさを0.4倍していることにも注意しよう。

V おわりに

本稿は、日経 NEEDS 財務データ MT 版にもとづいて金融関連会社に関するデータベースの構築を行い、その利用法に関して述べてきた。今回の構築によって、地道（2010-a）で課題として指摘されていた金融関連会社の財務データ等の抽出サービスを充実させることがある意味で実現したことになる。一方、今後データベースを迅速かつ正確に構築したり、サービスの拡大を行うのであれば、構築過程を自動化するスクリプトを開発する必要性が新たな課題として認識されたことも付記しておきたい。この課題の実現のためには Unix 系 OS で標準的なツールである make を利用したものがその候補として考えられる。なお、データ抽出環境はある程度整ってきたと考えてよいので、これらのデータベースを利用した非線形モデルにもとづく実証分析等が今後のメインの課題となるであろう。

（筆者は関西学院大学商学部教授）

謝辞

本稿で扱ったデータベースの構築は、地道（2010-a, b）における環境がなければ実現できなかったことから、そのシステムの構築に際して得た関西学院大学・商学部・産業研究所からの資金面での援助、およびシステム構築に関する有益なコメントをいただいた関西学院大学経済学部豊原法彦教授に対して心から感謝の意を表したい。

16) lattice パッケージに付属のプロット関数は複数の矩形パネルが格子状に配列された形状をもち、各パネルにはそこにプロットされる変数名などを表示する「帯」状の領域があり、ストリップ (strip) とよばれる。

参考文献

- [1] P. デュボワ (赤井誠他訳) (2003) 『MySQL クックブック VOLUME 1, 2』, オライリー・ジャパン.
- [2] 地道正行 (2010-a) 『日経 NEEDS 財務データにもとづくデータベースサーバの構築』, 商学論究, 第57巻, 第4号, pp.23-80.
- [3] 地道正行 (2010-b) 『財務データベースサーバの構築』, 関西学院大学レポジトリ, <http://kgur.kwansei.ac.jp/dspace/handle/10236/6013>, ISBN:9784990553005.
- [4] 増永良文 (2003) 『リレーショナルデータベース入門 [新訂版]』, サイエンス社.
- [5] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ一般事業会社レイアウト Ver.4.19』, 日経メディアマーケティング株式会社.
- [6] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ一般事業会社項目定義書 Ver.4.31』, 日経メディアマーケティング株式会社.
- [7] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ銀行レイアウト Ver.4.19』, 日経メディアマーケティング株式会社.
- [8] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ銀行項目定義書 Ver.4.17』, 日経メディアマーケティング株式会社.
- [9] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ証券レイアウト Ver.4.15』, 日経メディアマーケティング株式会社.
- [10] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ証券項目定義書 Ver.4.14』, 日経メディアマーケティング株式会社.
- [11] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ保険レイアウト Ver.4.14』, 日経メディアマーケティング株式会社.
- [12] 日本経済新聞社デジタルメディア編 (2010) 『NEEDS 財務データ保険項目定義書 Ver.4.14』, 日経メディアマーケティング株式会社.
- [13] 西沢夢路 (2007) 『基礎からの MySQL』, SoftBank Creative.
- [14] D. ショーカー (石田基広, 石田和枝共訳) (2009) 『R グラフィックス自由自在』, シュプリンガー・ジャパン.
- [15] P. スペクター (石田基広, 石田和枝 共訳) (2008) 『R データ自由自在』, シュプリンガー・ジャパン.
- [16] R Development Core Team (2011) *R Data Import/Export*, URL <http://www.r-project.org/>, R Foundaition for Statistical Computing, Vienna, Austria, ISBN 3-900051-10-0.
- [17] 豊原法彦 (2004) 『地域総合データ読出しシステムの構築』, 経済学論究, 第58巻3号, pp 159-175.
- [18] 豊原法彦 (2009) 『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』, 経済学論究, 第63巻第3号, pp 165-177.

付録

A 日経 NEEDS 財務データ MT 版に関する解説

A.1 提供データ

本稿で利用したデータセットは、日経メディアマーケティング株式会社から提供されている「日経 NEEDS 財務データ MT 版」である。収録されているデータは「財務確報」に関する「決算期データ」であり、全国上場会社（ヘラクレス、マザーズ含む、外国部除く）、ジャスダック上場会社、非上場有価証券報告書提出会社等の財務関連の情報が表4～7に与えられているファイル（gzip 形式）に納められている。

表4：日経 NEEDS データ MT 版ファイル一覧：一般事業会社

ファイル名	決算種別	説明
HIFSCA052.1970.gz	連結本決算	1970年代のデータファイル（圧縮済）
HIFSCA052.1980.gz	連結本決算	1980年代のデータファイル（圧縮済）
HIFSCA052.1990.gz	連結本決算	1990年代のデータファイル（圧縮済）
HIFSCA052.2000.gz	連結本決算	2000年代のデータファイル（圧縮済）
HIFSCA052.2010.gz	連結本決算	2010年代のデータファイル（圧縮済）
HIFSUA052.1960.gz	単独本決算	1960年代のデータファイル（圧縮済）
HIFSUA052.1970.gz	単独本決算	1970年代のデータファイル（圧縮済）
HIFSUA052.1980.gz	単独本決算	1980年代のデータファイル（圧縮済）
HIFSUA052.1990.gz	単独本決算	1990年代のデータファイル（圧縮済）
HIFSUA052.2000.gz	単独本決算	2000年代のデータファイル（圧縮済）
HIFSUA052.2010.gz	単独本決算	2010年代のデータファイル（圧縮済）

表5：日経 NEEDS データ MT 版ファイル一覧：銀行

ファイル名	決算種別	説明
HBFSCA050.2000.gz	連結本決算	2000年代のデータファイル（圧縮済）
HBFSCA050.2010.gz	連結本決算	2010年代のデータファイル（圧縮済）
HBFSUA050.1970.gz	単独本決算	1970年代のデータファイル（圧縮済）
HBFSUA050.1980.gz	単独本決算	1980年代のデータファイル（圧縮済）
HBFSUA050.1990.gz	単独本決算	1990年代のデータファイル（圧縮済）
HBFSUA050.2000.gz	単独本決算	2000年代のデータファイル（圧縮済）
HBFSUA050.2010.gz	単独本決算	2010年代のデータファイル（圧縮済）

表6：日経 NEEDS データ MT 版ファイル一覧：証券

ファイル名	決算種別	説明
HSFSCA050.2000.gz	連結本決算	2000年代のデータファイル（圧縮済）
HSFSCA050.2010.gz	連結本決算	2010年代のデータファイル（圧縮済）
HSF SUA050.1980.gz	単独本決算	1980年代のデータファイル（圧縮済）
HSF SUA050.1990.gz	単独本決算	1990年代のデータファイル（圧縮済）
HSF SUA050.2000.gz	単独本決算	2000年代のデータファイル（圧縮済）
HSF SUA050.2010.gz	単独本決算	2010年代のデータファイル（圧縮済）

表7：日経 NEEDS データ MT 版ファイル一覧：保険

ファイル名	決算種別	説明
HNFSCA050.2000.gz	連結本決算	2000年代のデータファイル（圧縮済）
HNFSCA050.2010.gz	連結本決算	2010年代のデータファイル（圧縮済）
HNFSUA050.1980.gz	単独本決算	1980年代のデータファイル（圧縮済）
HNFSUA050.1990.gz	単独本決算	1990年代のデータファイル（圧縮済）
HNFSUA050.2000.gz	単独本決算	2000年代のデータファイル（圧縮済）
HNFSUA050.2010.gz	単独本決算	2010年代のデータファイル（圧縮済）

これらのファイルを展開するとデータが格納されたテキストファイルが得られる。たとえば、2010年代の銀行に関する連結決算データに関する圧縮ファイル HBFSCA050.2010.gz を展開することによって、ファイル HBFSCA050.2010_27235.code が得られ、その先頭の1行には、以下のように2010年3月31日決算の「商工組合中央金庫」のデータが納められている。

なお、-99999999999999999999 は「欠損値」を表すことに注意しよう。これらのデータの配列等の詳細は、日経 NEEDS 財務データ MT 版に付属のマニュアル [5]～[12] を参照されたい。

B データベースの構造

構築された日経 NEEDS 財務データベースは以下のようなものである：

表9：データベースとテーブルに関するコード

業種	連結本決算	単独本決算	テーブルコード
一般事業会社	hifscsa052	hifsua052	fa01～fg01
銀行	hbfsca050	hbfsua050	fa11～fg11
証券会社	hsfscsa050	hsfsua050	fa21～fg21
保険会社	hnfscsa050	hnfsua050	fa31～fg31

ここで、各データベースには共通のテーブル `firmlist` があり、各テーブルコードは以下のようなことを意味する。

表10：テーブル一覧

テーブル名	内容
fa*1	属性
fb*1	貸借対照表
fc*1	損益計算書
fd*1	キャッシュフロー計算書
fe*1	その他
ff*1	その他 2
fg*1	株主資本等変動計算書
firmlist	収録企業リスト

テーブル `fx*1` ($x=a, b, \dots, g, * = 0, 1, 2, 3$) のカラム (列) は以下のような共通の構造を持っていることに注意しよう：

$$\begin{aligned} \text{テーブル} &= \text{ヘッダー部} + \text{データ部} \\ \text{fx*1} &= \text{a01} \sim \text{a32} + \text{b001} \sim \text{b210} \end{aligned}$$

ここで、ヘッダー部は、テーブル $f_x \times 1$ に関して共通であり、カラムの詳細は表12で与えられる。また、各テーブル $f_x \times 1$ に関するデータ部の詳細は表13～19を参照されたい¹⁷⁾。さらに、テーブル `firmlist` は、表11で与えられるようなカラムを持つ収録企業のリストである。

表11：テーブル `firmlist` のカラム

カラム名	内容	データ型
<code>nikkeicode</code>	日経会社コード	VARCHAR(7)
<code>stockcode</code>	株式コード	VARCHAR(4)
<code>nikkeigyousyucode</code>	日経業種コード	VARCHAR(6)
<code>shamei en</code>	社名 (英文表記)	VARCHAR(42)
<code>shamei jp</code>	社名 (和文表記)	VARCHAR(42)
<code>shamei jpk</code>	社名 (カタカナ表記)	VARCHAR(56)

表12：ヘッダー詳細 (単独, 連結, `fall`～`fg11` 共通)

項番	位置	桁数	カラム名	データ型	項目名	内容
1	1	4	<code>a01</code>	VARCHAR(4)	レコード種別 #: E:財務速報, F:確報 *: 一般事業会社:0, 銀行:1, 証券:2, 保険:3, 信用金庫:4	#A*1:属性 #B*1:貸借対照表 #C*1:損益計算書 #D*1:キャッシュフロー表 #E*1:注記・明細1 (有価証券注 記, 退職給付会計等) #F*1:注記・明細2 (リース会計, デリバティブ注記等) #G*1:株主資本等変動計算書
2	5	8	<code>a02</code>	DATE	決算年月日	YYYYMMDD
3	13	1	<code>a03</code>	VARCHAR(1)	識別フラグ	新規・修正: 1, 削除: 9
4	14	7	<code>a04</code>	VARCHAR(7)	日経会社コード	日経が定める会社コード
5	21	4	<code>a05</code>	VARCHAR(4)	株式コード	証券コード協議会が定める4桁 の会社コード
6	25	5	<code>a06</code>	VARCHAR(5)	予備	新株式コードのための予備 (ス ベース)
7	30	2	<code>a07</code>	VARCHAR(2)	決算月数	決算月数

17) 紙面の都合上、これらの表の項目の一部を省略していることに注意されたい。なお、すべての項目を含むリストのファイル (PDF, HTML 形式) は、データ抽出システム KGUSBADES のトップページからダウンロード可能であることに注意しよう。また、すべての項目とその詳細は、日経 NEEDS 財務データ MT 版に付属のマニュアル [5]～[12]を参照されたい。

項番	位置	桁数	カラム名	データ型	項目名	内容
8	32	1	a08	VARCHAR(1)	連結・単独フラグ	単独: 1, 連結: 2
9	33	1	a09	VARCHAR(1)	本・中間決算フラグ	本決算: 1 中間決算: 2 9ヶ月累計四半期: 3 第1四半期: 4 第2四半期: 5 第3四半期: 6 第4四半期: 7
10	34	1	a10	VARCHAR(1)	決算期フラグ	新規収録分: 1, 次期における修正分: 0
11	35	8	a11	VARCHAR(8)	発表決算期	このデータが発表された決算期 YYYYMM00
12	43	17	a12	VARCHAR(17)	予備	スペース
13	60	1	a13	VARCHAR(1)	連結基準フラグ	日本基準: 1, SEC基準: 2 (単独時は0)
14	61	1	a14	VARCHAR(1)	上場フラグ	上場中: 1, 未上場・上場廃止: 0
15	62	2	a15	VARCHAR(2)	上場場部	東証1部: 11 東証マザーズ: 13 東証2部: 12 ヘラクレス 大証1部: 21 スタンダード: 91 大証2部: 22 グロース : 94 名証1部: 31 名証2部: 32 京都: 41 広島: 51 福岡: 61 新潟: 71 札幌: 81 未上場: 99 ※上場廃止会社は、廃止時の場部
16	64	1	a16	VARCHAR(1)	ジャスダックフラグ	ジャスダック上場: 1, ジャスダック未上場: 0
17	65	2	a17	VARCHAR(2)	ジャスダック市場	ジャスダック上場: 11 ジャスダック未上場: 99 ※上場廃止会社は、廃止時のジャスダック市場
18	67	1	a18	VARCHAR(1)	有報フラグ	有報提出会社: 1, 非有報提出会社: 0
19	68	1	a19	VARCHAR(1)	未上場収録フラグ	未上場会社として収録を継続している: 1 収録廃止会社, 上場会社: 0
20	69	5	a20	VARCHAR(5)	予備	スペース
21	74	8	a21	DATE	データ作成日	YYYYMMDD
22	82	6	a22	VARCHAR(6)	日経業種コード	ABBCCC A...製造業: 1, 非製造業: 2B...日経業種中分類コード C...日経業種小分類コード
23	88	1	a23	VARCHAR(1)	上場情報: 東京	1部上場: 1, 2部上場: 2, マザーズ: 3, 未上場: 0
24	89	1	a24	VARCHAR(1)	〃 : 大阪	1部上場: 1, 2部上場: 2, 未上場: 0
25	90	1	a25	VARCHAR(1)	〃 : 名古屋	1部上場: 1, 2部上場: 2, 未上場: 0
26	91	1	a26	VARCHAR(1)	〃 : 京都	1部上場: 1, 未上場: 0
27	92	1	a27	VARCHAR(1)	〃 : 広島	1部上場: 1, 未上場: 0
28	93	1	a28	VARCHAR(1)	〃 : 福岡	1部上場: 1, 未上場: 0

項番	位置	桁数	カラム名	データ型	項目名	内容
29	94	1	a29	VARCHAR(1)	々 : 新潟	1 部上場: 1, 未上場: 0
30	95	1	a30	VARCHAR(1)	々 : 札幌	1 部上場: 1, 未上場: 0
31	96	1	a31	VARCHAR(1)	々 : ヘラクレス	スタンダード: 1, グロース: 4
32	97	4	a32	VARCHAR(4)	予備	スペース

表13：属性 (fa11)：データ部 (単独, 連結 共通)

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	レコード FA11/EA11 収録フラグ
2	14	b002	VARCHAR(14)	レコード FB11/EB11 収録フラグ
3	14	b003	VARCHAR(14)	レコード FC11/EC11 収録フラグ
4	14	b004	VARCHAR(14)	レコード FD11/ED11 収録フラグ
5	14	b005	VARCHAR(14)	レコード FE11/EE11 収録フラグ
6	14	b006	VARCHAR(14)	レコード FF11/EF11 収録フラグ
7	14	b007	VARCHAR(14)	レコード FG11/EG11 収録フラグ
8	14	b008	VARCHAR(14)	予備
9	14	b009	VARCHAR(14)	予備
10	14	b010	VARCHAR(14)	予備
(中略):	:	:	:	:
210	14	b210	VARCHAR(14)	予備

表14：貸借対照表 (fb11)：データ部 (単独, 連結 共通)

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	現金預け金合計
2	14	b002	VARCHAR(14)	現金
3	14	b003	VARCHAR(14)	(うち切手手形)
4	14	b004	VARCHAR(14)	預け金
5	14	b005	VARCHAR(14)	(うち日銀への預け金)
6	14	b006	VARCHAR(14)	コールローン及び買入手形
7	14	b007	VARCHAR(14)	コールローン
8	14	b008	VARCHAR(14)	買入手形
9	14	b009	VARCHAR(14)	買現先勘定
10	14	b010	VARCHAR(14)	買入金銭債権
(中略):	:	:	:	:
210	14	b210	VARCHAR(14)	予備

表15：損益計算書（fc11）：データ部（単独，連結 共通）

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	経常収益合計
2	14	b002	VARCHAR(14)	資金運用収益合計
3	14	b003	VARCHAR(14)	貸出金利息
4	14	b004	VARCHAR(14)	(うち貸付金利息)
5	14	b005	VARCHAR(14)	(うち手形割引料)
6	14	b006	VARCHAR(14)	有価証券利息配当金
7	14	b007	VARCHAR(14)	(うち株式配当金)
8	14	b008	VARCHAR(14)	コールローン利息及び買入手形利息
9	14	b009	VARCHAR(14)	コールローン利息
10	14	b010	VARCHAR(14)	買入手形利息
(中略)：	：	：	：	：
210	14	b210	VARCHAR(14)	予備

表16：キャッシュフロー計算書（fd11）：データ部（単独，連結 共通）

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	税金等調整前当期純利益
2	14	b002	VARCHAR(14)	繰延税金
3	14	b003	VARCHAR(14)	減価償却費
4	14	b004	VARCHAR(14)	連結調整勘定償却額
5	14	b005	VARCHAR(14)	持分法による投資損益（△益）
6	14	b006	VARCHAR(14)	貸倒引当金の増加額（△減少額）
7	14	b007	VARCHAR(14)	投資損失引当金の増加額（△減少額）
8	14	b008	VARCHAR(14)	債権売却損失引当金の増加額（△減少額）
9	14	b009	VARCHAR(14)	退職給付引当金の増加額（△減少額）
10	14	b010	VARCHAR(14)	その他の引当金増加額
10	14	b010	VARCHAR(14)	買入手形利息
(中略)：	：	：	：	：
210	14	b210	VARCHAR(14)	予備

表17：その他（fe11）：データ部（単独，連結 共通）

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	国際業務経常収益
2	14	b002	VARCHAR(14)	共同債権買取機構向け債権売却損
3	14	b003	VARCHAR(14)	累積債務国向け債権売却損失
4	14	b004	VARCHAR(14)	有証評価方法フラグ（低：1，原：2）
5	14	b005	VARCHAR(14)	減価償却方法フラグ（率：1，額：2）
6	14	b006	VARCHAR(14)	有形固定資産減価償却累計額
7	14	b007	VARCHAR(14)	動産・不動産の減価償却実施額
8	14	b008	VARCHAR(14)	有形固定資産の圧縮記帳額
9	14	b009	VARCHAR(14)	自己株式
10	14	b010	VARCHAR(14)	担保に供している資産合計
(中略)：	：	：	：	：
210	14	b210	VARCHAR(14)	うちオペレーショナルリスク相当額(国内基準)

表18：その他2（ff11）：データ部（単独，連結 共通）

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	破綻先債権額 (銀行勘定)
2	14	b002	VARCHAR(14)	延滞債権額 (銀行勘定)
3	14	b003	VARCHAR(14)	金利減免等債権額 (銀行勘定)
4	14	b004	VARCHAR(14)	経営支援先債権額 (銀行勘定)
5	14	b005	VARCHAR(14)	3ヶ月以上延滞債権額 (銀行勘定)
6	14	b006	VARCHAR(14)	貸出条件緩和債権額 (銀行勘定)
7	14	b007	VARCHAR(14)	破綻先債権額 (信託勘定)
8	14	b008	VARCHAR(14)	延滞債権額 (信託勘定)
9	14	b009	VARCHAR(14)	3ヶ月以上延滞債権額 (信託勘定)
10	14	b010	VARCHAR(14)	貸出条件緩和債権額 (信託勘定)
(中略)：	：	：	：	：
210	14	b210	VARCHAR(14)	予備

表19：株主資本等変動計算書（fg11）：データ部（単独，連結 共通）

項番	桁数	カラム名	データ型	項目名
1	14	b001	VARCHAR(14)	資本金：前期未残高
2	14	b002	VARCHAR(14)	資本金：新株の発行
3	14	b003	VARCHAR(14)	予備
4	14	b004	VARCHAR(14)	予備
5	14	b005	VARCHAR(14)	予備
6	14	b006	VARCHAR(14)	予備
7	14	b007	VARCHAR(14)	予備
8	14	b008	VARCHAR(14)	資本金：資本金から準備金または剰余金への振替
9	14	b009	VARCHAR(14)	資本金：準備金から資本金または剰余金への振替
10	14	b010	VARCHAR(14)	資本金：剰余金から資本金または準備金への振替
(中略)：	：	：	：	：
210	14	b210	VARCHAR(14)	予備

C SQL 問合せの例

典型的な SQL 問合せの例を以下に与える。

(SQL-1) 収録されている全銀行リストを抽出

```
SELECT * FROM firmlist;
```

[解説] 社名や日経会社コードが納められているテーブル `firmlist` からすべてのカラム (*) を抽出する (SELECT) ことを意味する。

(SQL-2) みずほ銀行の「決算年月日」, 「決算月数」, 「預金合計」を抽出 (決

算年月日順)

```
SELECT a02,a07,b070 FROM fb11 WHERE a04='0070004' ORDER BY a02;
```

[解説] 貸借対照表のテーブル fb11 におけるカラム a04 (日経会社コード) において 0070004 (みずほ銀行の日経会社コード) という文字列を含むフィールドを a02 (決算年月日), a07 (決算月数), b070 (預金合計) に対して, 決算年月日順 (ORDER BY a02) に抽出する (SELECT) ことを意味する.

(SQL-3) 2010年3月31日決算の東京証券取引所1部上場銀行の「日経会社コード」と「経常収益合計」, 「経常利益」の抽出 (日経会社コード順)

```
SELECT a04,b001,b090 FROM fc11
      WHERE a02='2010-03-31' AND a23='1' ORDER BY a04;
```

[解説] 損益計算書のテーブル fc11 におけるカラム a02 (決算年月日) において 2010-03-31 (2010年3月31日) であり, かつカラム a23 (上場情報:東京) において 1 (1部上場) であるフィールドを a04 (日経会社コード), b001 (経常収益合計), b090 (経常利益) に対して, 日経会社コード順 (ORDER BY a04) に抽出する (SELECT) ことを意味する.

(SQL-4) 2010年3月31日決算の東京証券取引所1部上場銀行の「日経会社コード」, 「銀行名」, 「経常収益合計」, 「経常利益」の抽出 (日経会社コード順)

```
SELECT fc11.a04,firmlist.shamei_jp,fc11.b001,fc11.b090 FROM fc11
      JOIN firmlist ON fc11.a04=firmlist.nikkeicode
      WHERE a02='2010-03-31' AND a23='1' ORDER BY a04;
```

[解説] 損益計算書のテーブル fc11 におけるカラム a02 (決算年月日) において 2010-03-31 (2010年3月31日) であり, かつカラム a23 (上場情報:東京) において 1 (1部上場) であるレコー

ドに対して損益計算書のテーブルにおける日経会社コードのカラム (fc01.a04) と企業 (銀行) リストのテーブルにおける日経会社コードのカラム (firmlist.nikkeicode) とを結合 (JOIN) した結果から, fc11.a04 (損益計算書のテーブルにおける日経会社コードのカラム), firmlist.shamei_jp (企業リストのテーブルにおける日本語の銀行名), fc11.b001 (損益計算書のテーブルにおける経常収益合計) と fc11.b090 (損益計算書のテーブルにおける経常利益) に対して日経会社コード順 (ORDER BY a04) に抽出する (SELECT) ことを意味する.

(SQL-5) 東京証券取引所 1 部上場銀行の「日経会社コード」, 「銀行名」, 「決算年月日」, 「経常収益合計」, 「経常利益」のパネルデータ抽出 (日経会社コード, 決算年月日順)

```
SELECT fc11.a04, firmlist.shamei_jp, fc11.a02, fc11.b001, fc11.b090 FROM fc11
      JOIN firmlist ON fc11.a04=firmlist.nikkeicode
      WHERE a23='1' ORDER BY a04, a02;
```

[解説] 損益計算書のテーブル fc11 におけるカラム a23 (上場情報: 東京) において 1 (1 部上場) であるレコードに対して損益計算書のテーブルにおける日経会社コードのカラム (fc11.a04) と銀行リストのテーブルにおける日経会社コードのカラム (firmlist.nikkeicode) とを結合 (JOIN) した結果から, fc11.a04 (損益計算書のテーブルにおける日経会社コードのカラム), firmlist.shamei_jp (銀行リストのテーブルにおける日本語の銀行名), fc01.b001 (損益計算書のテーブルにおける経常収益合計) と fc01.b006 (損益計算書のテーブルにおける経常利益) に対して日経会社コードと決算年月日順 (ORDER BY a04, a02) に抽出する (SELECT) ことを意味する.

D データベース構築に関するスクリプトファイル

D.1 OS 共通スクリプトファイル

ファイル loadtoMySQL.sql

```
DROP DATABASE IF EXISTS hbfsca050;
CREATE DATABASE hbfsca050;
USE hbfsca050;
DROP TABLE IF EXISTS fall;
DROP TABLE IF EXISTS fb11;
DROP TABLE IF EXISTS fc11;
DROP TABLE IF EXISTS fd11;
DROP TABLE IF EXISTS fe11;
DROP TABLE IF EXISTS ff11;
DROP TABLE IF EXISTS fg11;
CREATE TABLE fall (
a01 VARCHAR(4),
a02 DATE,
a03 VARCHAR(1),
(中略)
a32 VARCHAR(4),
b011 VARCHAR(14),
b002 VARCHAR(14),
b003 VARCHAR(14),
(中略)
b210 VARCHAR(14)
);
CREATE TABLE fb11 LIKE fall;
CREATE TABLE fc11 LIKE fall;
CREATE TABLE fd11 LIKE fall;
CREATE TABLE fe11 LIKE fall;
CREATE TABLE ff11 LIKE fall;
CREATE TABLE fg11 LIKE fall;
LOAD DATA INFILE './FAlltab.data' INTO TABLE fall FIELDS TERMINATED BY '\t';
LOAD DATA INFILE './FB11tab.data' INTO TABLE fb11 FIELDS TERMINATED BY '\t';
LOAD DATA INFILE './FC11tab.data' INTO TABLE fc11 FIELDS TERMINATED BY '\t';
LOAD DATA INFILE './FD11tab.data' INTO TABLE fd11 FIELDS TERMINATED BY '\t';
LOAD DATA INFILE './FE11tab.data' INTO TABLE fe11 FIELDS TERMINATED BY '\t';
LOAD DATA INFILE './FF11tab.data' INTO TABLE ff11 FIELDS TERMINATED BY '\t';
LOAD DATA INFILE './FG11tab.data' INTO TABLE fg11 FIELDS TERMINATED BY '\t';
```

ファイル loadfirmlisttoMySQL.sql

```
USE hbfsca050;
DROP TABLE IF EXISTS firmlist;
CREATE TABLE firmlist (
nikkeicode VARCHAR(7),
stockcode VARCHAR(4),
nikkeigyousyucode VARCHAR(6),
shamei_en VARCHAR(42),
shamei_jp VARCHAR(42),
shamei_jpk VARCHAR(56)
);
LOAD DATA INFILE './firmlistbank.txt' INTO TABLE firmlist FIELDS TERMINATED BY '\t';
```

D.2 Mac OS X Server, Ubuntu 用スクリプトファイル

ファイル script.sh

```

echo Start unzip!
#
gunzip -c HIFSCA050.2000.gz > renketsuall.txt
gunzip -c HBFSCA050.2010.gz >> renketsuall.txt
#
echo Complete unzip!
#
grep FA11 renketsuall.txt | sed '/^$/d' > FA.data
grep FB11 renketsuall.txt | sed '/^$/d' > FB.data
grep FC11 renketsuall.txt | sed '/^$/d' > FC.data
grep FD11 renketsuall.txt | sed '/^$/d' > FD.data
grep FE11 renketsuall.txt | sed '/^$/d' > FE.data
grep FF11 renketsuall.txt | sed '/^$/d' > FF.data
grep FG11 renketsuall.txt | sed '/^$/d' > FG.data
#
echo Complete Separate!
#
perl -l separate.pl FA.data; sed '/^$/d' temp | nkf --utf8 > FAtab.data; rm temp;
perl -l separate.pl FB.data; sed '/^$/d' temp | nkf --utf8 > FBtab.data; rm temp;
perl -l separate.pl FC.data; sed '/^$/d' temp | nkf --utf8 > FCtab.data; rm temp;
perl -l separate.pl FD.data; sed '/^$/d' temp | nkf --utf8 > FDtab.data; rm temp;
perl -l separate.pl FE.data; sed '/^$/d' temp | nkf --utf8 > FEtab.data; rm temp;
perl -l separate.pl FF.data; sed '/^$/d' temp | nkf --utf8 > FFtab.data; rm temp;
perl -l separate.pl FG.data; sed '/^$/d' temp | nkf --utf8 > FGtab.data; rm temp;
#
echo Complete Tables!
#
rm renketsuall.txt
rm FA.data FB.data FC.data FD.data FE.data FF.data FG.data
#
echo Finish!

```

ファイル separate.pl

```

# code ファイルを読み込んで、項目ごとの列に分割する。
# セパラータはタブ(\t) とする。
#####
open(IN, "@ARGV");
open(OUT, ">temp");
#####
do {
  read(IN, $xx, 3042);
  $a[1]=substr($xx, 0, 4);
  $a[2]=substr($xx, 4, 8);
  $a[3]=substr($xx, 12, 1);
  (中略)
  $a[32]=substr($xx, 96, 4);
  $b[1]=substr($xx, 100, 14);
  $b[2]=substr($xx, 114, 14);
  $b[3]=substr($xx, 128, 14);
  (中略)
  $b[210]=substr($xx, 3026, 14);
  print OUT
  "$a[1]\t$a[2]\t$a[3]\t$a[4]\t$a[5]\t (中略) \t$a[32]\t
  $b[1]\t$b[2]\t$b[3]\t$b[4]\t$b[5]\t (中略) $b[209]\t$b[210]\n";
} while (eof(IN)!=1);
close (IN);

```

D.3 Windows 用スクリプトファイル

ファイル script.sh

```

echo Start unzip!
gunzip -c HBFSCA050.2000.gz > renketsuall.txt
gunzip -c HBFSCA050.2010.gz >> renketsuall.txt
#
echo Complete unzip!
#
grep FA11 renketsuall.txt | sed '/^$/d' > FA.data
grep FB11 renketsuall.txt | sed '/^$/d' > FB.data
grep FC11 renketsuall.txt | sed '/^$/d' > FC.data
grep FD11 renketsuall.txt | sed '/^$/d' > FD.data
grep FE11 renketsuall.txt | sed '/^$/d' > FE.data
grep FF11 renketsuall.txt | sed '/^$/d' > FF.data
grep FG11 renketsuall.txt | sed '/^$/d' > FG.data
#
echo Complete Separate!
#
perl -l separate.pl FA.data; sed '/^$/d' temp > FAtab.data; rm temp;
perl -l separate.pl FB.data; sed '/^$/d' temp > FBtab.data; rm temp;
perl -l separate.pl FC.data; sed '/^$/d' temp > FCtab.data; rm temp;
perl -l separate.pl FD.data; sed '/^$/d' temp > FDtab.data; rm temp;
perl -l separate.pl FE.data; sed '/^$/d' temp > FEtab.data; rm temp;
perl -l separate.pl FF.data; sed '/^$/d' temp > FFtab.data; rm temp;
perl -l separate.pl FG.data; sed '/^$/d' temp > FGtab.data; rm temp;
#
echo Complete Tables!
#
rm renketsuall.txt
rm FA.data FB.data FC.data FD.data FE.data FF.data FG.data
#
echo Finish!

```

ファイル separate.pl

```

# code ファイルを読み込んで、項目ごとの列に分割する。
# セパレータはタブ (\t) とする。
#####
open(IN, "@ARGV");
open(OUT, ">temp");
#####
do{
  read(IN, $xx, 3041);
  $a[1]=substr($xx, 0, 4);
  $a[2]=substr($xx, 4, 8);
  $a[3]=substr($xx, 12, 1);
  (中略)
  $a[32]=substr($xx, 96, 4);
  $b[1]=substr($xx, 100, 14);
  $b[2]=substr($xx, 114, 14);
  $b[3]=substr($xx, 128, 14);
  (中略)
  $b[210]=substr($xx, 3026, 14);
  print OUT
  "$a[1]\t$a[2]\t$a[3]\t$a[4]\t$a[5]\t (中略) \t$a[32]\t
  $b[1]\t$b[2]\t$b[3]\t$b[4]\t$b[5]\t (中略) $b[209]\t$b[210]\n";
} while (eof(IN)!=1);
close (IN);

```
