

# WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

## 1. 1. はじめに

今まで、筆者は、Windows環境でUNIXライクなコマンドを使いたい、UNIX形式でのみコンパイルされたソフト（例えば、PostgreSQL）を使いたい、といった事を考えていて、どうすればそれが可能なのか、を色々と探してきました。特に、PostgreSQLは、ローカル環境で動かしたい、という強い要求がありました。

数年前（といっても、もう8年近く前になります）は、BSD on Windows (BOW) というソフトも使いましたが、どうも使い勝手がよいようには思えず、また、メモリを多量に使うため、BOWを立ち上げるだけでWindows（当時は95）がフリーズする現象が起き、実際にはあまり使えませんでした。

このドキュメントでは、Cygwinの導入の方法と、入れる際の注意点、PostgreSQLのCygwin版の導入方法などを綴っていきます。

ただ、最近では、[VMware](#)を利用したOSのエミュレーションの方が、パフォーマンスにおいても良いし制約も無くてよいか、と思っています。Windows OS上でLinuxやFreeBSDなどがエミュレートできます。想定できるソリューションも多数あります。また、Java派の方は、[「Jakartaの杜」](#)を参照するのも良いでしょう。ほぼ、Windows上で、UNIXと同じ事がJavaで出来るかと思っております。

Windows2000、Windows98での確認をしております。尚、Win98では、postgreSQLは動きません。（安定してません）

## 2. 2. Cygwinとは？

Cygwin ツール群は、広く普及している GNU の開発ツールを、Windows NT(2000)、95、98 用に移植したものです。これらのプログラムは、それらが必要としている UNIX システムコールや環境を提供する Cygwin ライブラリによって動作します。標準では、bashシェルで動かします。Cygwinというところが開発していたのですが、[REDHAT](#)社に買収されました。[Apache Jakarta Project](#) の方々は、何故か、「Windows環境でCVSを使うにはCygwinを。。。。」と言っているようです。（勿論、

Windows環境の場合、CVSにはWInCVSを使っても良いです)

### 3. 3. Cygwinが提供するもの

フルリリースには以下のパッケージが含まれます。

・開発ツール: binutils, bison, byacc, dejagnu, diff, expect, flex, gas, gcc, gdb, itcl, ld, libstdc++, make, patch, tcl, tix, tk

ユーザツール: ash, bash, bzip2, diff, fileutils, findutils, gawk, grep, gzip, m4, sed, shellutils, tar, textutils, time

ユーザツールリリースはユーザツールしか含んでいません。

これらのツールの全てのソースコードが利用可能です。

### 4. 4. ダウンロード方法

以下のいずれかのサイト (FTP) に行き、その中の「setup.exe」をダウンロードしてください。現在は、バージョン1.3.10のはず・・・

[ftp.u-aizu.ac.jp](ftp://u-aizu.ac.jp)

[ring.aist.go.jp](http://ring.aist.go.jp)

[ring.crl.go.jp](http://ring.crl.go.jp)

[ring.etl.go.jp](http://ring.etl.go.jp)

上でダウンロードしたsetup.exeを起動します。起動すると、

- ・ Download from Internet
- ・ Install from Internet
- ・ Install from Local Directory

の三種類が選べますが、最初は、「Download from Internet」を選択しましょう。

(Install from Internetでも良いのですが、ダウンロード途中で接続が切れた場合などのことを考えて・・・)

次に、Local Package Directoryを聞いてきますので、Cygwinファイル群 (バイナリ・ソース) をダウンロードする場所を入力します。例えば、「C:\Temp\cygwin」などといった感じです。

接続形式は、「Use IE Settings」「Direct Connection」のどちらかで良いでしょう。Proxyサーバーを使う方は、「Use HTTP/FTP Proxy」を選択してください。

## WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

Nextボタンを押すと、ダウンロード出来るFTPサイトの一覧が表示されます。例えば、<ftp://ftp.etl.go.jp/>のように、日本のサイトを選択すると良いでしょう。setup.iniがダウンロードされます。

インストールするパッケージの一覧が出てきます。そのまま「->Next」を押しても良いでしょう（注：Skipと表示されているところで、マウスの左クリックをすると、最新バージョンを得ることができます。）。ダウンロードが開始されます。尚、ソースもダウンロードすると、かなりの時間がかかります。

ダウンロードが終了すれば、「Download Complete」と出ます。

### 5. 5. インストール方法

先ほどの「setup.exe」をもう一度起動します。

- ・ Download from Internet
- ・ Install from Internet
- ・ Install from Local Directory

の三種類が選べますが、今度は、「Install from Local Directory」を選択しましょう。

次に、Local Package Directoryを聞いてきますので、Cygwinファイル群（バイナリ・ソース）をダウンロードした場所を入力します。例えば、「C:\Temp\cygwin」などといった感じです。

更に、「Select Install Directory」と出てきます。ここで、実際にインストールを行うディレクトリを指定します。例えば、「C:\Cygwin」といった具合です。Default Text File TypeはUNIX、Install forはALLをチェックし、次にすすみます。インストールするパッケージは、そのままにして良いでしょう。「->Next」を押します。

多少時間がかかりますが、インストールが終了すると、「Create Desktop Icon」「Add to start menu」と出てきます。両方ともチェックを入れて「->Next」を押せば、デスクトップにアイコンが作成され、スタートメニューにCygwinが登録されます。これでインストールは終了です。

### 6. 6. 設定方法

Cygwin ユーザーズガイドの日本語訳が[こちら](#)にあります。

特に、ディレクトリ構造の部分はしっかりとご覧ください。

[ここ](#)も役立ちます。(Apache設定するひともしないひとも)

あ・・・[ここ](#)もだ・・・。

私はダウンロード先を「C:\Cygwin\」にしましたが、お好きなディレクトリにしてかまいません。以後、%CYGDIR%と表記します。

NT系の場合、コントロールパネルの「環境変数」のPATHに、以下を追加します。

%CYGDIR%bin

コマンドプロンプトから、「bash」と打ってもよいですし、デスクトップにできた(であろう)アイコンをクリックしても良いです。これで、基本的な設定は完了です。

## 7. 7. PostgreSQLのCygwinへの導入

<必要ファイル>

Cygwin用の最新版のPostgreSQL (7.3.1) です。[cygwin1.3.10以降用](#)

尚、上記の「Cygwin導入」で、Postgresを入れた人は、<必要コマンド>以降から読んでください。

ダウンロードするディレクトリは、%CYGDIR%tmpがよいかと思います・・・。

ダウンロード後、ダウンロードしたディレクトリに行き、  
bash-2.03% bzip2 postgresql7.3.1-cygwin-mb.tar.bz2 | tar xvf - -C /usr/local

として、解凍します。(postgresql7.3.1-cygwin-mb.tar.bz2 の部分には、ダウンロードしたファイル名を入れます。適当に書き換えてください。)

/usr/local/pgsql/以下に、PostgreSQLがインストールされました。

次に、<http://www.nonsensecorner.com/pgsql/>にある[cygipc-1.13-2.tar.bz2](#)などをダウンロードします。

でてきた「exe」ファイルを、%CYGDIR%binなどの、パスの通った場所に入れます。

<必要コマンド>

作成するファイルは、/etc/passwd,/etc/groupsなどですが、これは以下のようなコマンドで作成してくれます。  
bash-2.03% mkpasswd -l > /etc/passwd  
bash-2.03% mkgroup -l > /etc/groups

WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

### <profileファイルの更新>

/etc/profileに、以下を追加します。

```
PGHOST=127.0.0.1
PATH=$PATH:/usr/local/pgsql/bin:/usr/local/bin:/usr/local/pgsql/lib
PGDATA=/usr/local/pgsql          PGLIB=/usr/local/pgsql/lib
LD_LIBRARY_PATH=/usr/local/pgsql/lib:/usr/local/lib
POSTGRES_LIB=/usr/local/pgsql/lib export LD_LIBRARY_PATH PGDATA PGLIB
POSTGRES_LIB if [ -f /usr/local/pgsql/postmaster.pid ]; then postid='ps ax|
grep postmaster |grep -v grep|awk '{print $1;}' ' if [ X${postid} != "X" ] ;
then kill -TERM ${postid} fi fi ipcid='ps ax| grep ipc-daemon |grep -v
grep|awk '{print $1;}' ' if [ X${ipcid} != "X" ] ; then kill -TERM ${ipcid}
fi ipc-daemon -q & nohup postmaster -i -p 5432 >logfile 2>&1 &
```

必要とあらば

```
PGCLIENTENCODING=SJIS export PGCLIENTENCODING
```

の2行を入れます。

下線が引いてある個所は、ipc-daemonのプロセスを確認し、あればKILLすること、Postmasterのプロセスを確認し、あればKILLすること、そして、その両方をたちあげること、をやっています。Cygwin立ち上げ時に自動的にipc-daemonとpostmasterをたちあげたい場合は入れるべきですが、自動化したくなくともよいです。

### <DB作成>

まず、initdbします。そのまえに、

```
bash-2.03% ipc-daemon -q &
```

しておきましょう。

```
bash-2.03% initdb -E EUC_JP
```

とします。(これでエラーが出る場合、データベース作成ディレクトリが指定されていないことが考えられます。その場合は、initdb -E EUC\_JP -D/usr/local/pgsql などとしてみてください。) ポイントは、文字コードをEUC\_JPにしていること。これがないと、Windows環境ですから、文字化けしてしまいます・・・。

#あと、Postgresの7.1.3のCygwinものから、(多分)デフォルトのコードがUNICODEになっているかと思われます。psql -lなどで確認ください。

initdbに成功すれば、指定されたディレクトリ以下に、データベースの雛型等ができます。例えば、/usr/local/pgsql以下にbaseというディレクトリができていますね？

POSTMASTERを起動します。

```
bash-2.03% postmaster -i &
```

上で、ipc-daemonを立ち上げていないと、shmget()エラーが出る場合があります。

さて、いよいよ、データベースの作成です。

```
bash-2.03% createdb -h 127.0.0.1 test
```

はい、これで、testというデータベースの出来上がりです。

あとは、

```
bash-2.03% psql -h 127.0.0.1 -n test
```

とすれば、テストデータベースに入ることができます。（なぜか、-h 127.0.0.1が必ず必要なのです・・・。Windows版のcreatedbやpsqlの仕様らしい・・・）

基本的に、postmasterが起動していればpsqlできますが、出来ない場合は、「postmasterをkillする」「ipc-daemonをkillする」「両方を立ち上げなおす」してみてください。kill出来ない場合は、Windowsのタスクマネージャーでプロセスをとめてしまいましょう。

以下に、PSQLを立ち上げた後の実際の例を示します。青色フォントが、実際に入力する文字です。黒色フォントは、DBサーバーからの応答です。

```
test=# create table tmptable (number serial not null primary key, value varchar(254));
NOTICE: CREATE TABLE will create implicit sequence 'tmptable_number_seq' for SERIAL
column 'tmptab
e.number'
NOTICE: CREATE TABLE/PRIMARY KEY will create implicit index 'tmptable_pkey' for table
'tmptable'
NOTICE: CREATE TABLE will create implicit sequence 'tmptable_number_seq' for SERIAL
column 'tmptab
e.number'
NOTICE: CREATE TABLE/PRIMARY KEY will create implicit index 'tmptable_pkey' for table
'tmptable'
CREATE
test=# insert into tmptable(value) values ('This is a test.');
```

```
INSERT 25087 1
```

```
test=# insert into tmptable(value) values ('日本語 テスト');
```

WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

```
INSERT 25088 1
test=# select * from tmptable;
number | value
-----+-----
1 | This is a test.
2 | 日本語テスト
(2 rows)
(END)
```

こうなれば、ばっちりですね。

PSQLを終了するとき、

```
test=#\q
```

です..

ちなみに、上で作ったテーブルを削除する場合は、

```
test=#drop sequence tmptable_number_seq;
DROP
test=# drop table tmptable;
DROP
```

#### 8. 8. 更なる最適環境構築の為に

<postgresql ODBC ドライバーの導入 (Windowsからリモートのpostgresqlなどと接続: MS-ACCESSなどで利用できる) >

[ftp://ftp.postgresql.org/pub/odbc/versions/full/psqlodbc-07\\_01\\_0009.zip](ftp://ftp.postgresql.org/pub/odbc/versions/full/psqlodbc-07_01_0009.zip)

から、オリジナルのPostgreSQL ODBC Driverを入手。

[psqlodbc-jp-20010713-bin.zip \(119,907バイト\) - zip で圧縮したもの](#)

から日本語パッチをダウンロードします。

(あるいは、[「PostgreSQL ODBC Driver 日本語版」](#)や、[「PostgreSQL ODBC Driver 日本語版のインストールについて」](#)をご参照ください。)

日本語版をインストールするにあたり、先にインストールするオリジナル (英語版) のバージョンは特に問いません。

オリジナル (英語版) に付属のODBCドライバマネージャは英語版です。インストール先のシステムにODBCドライバマネージャがないか、あってもバージョンが古い場合、オリジナルのセットアッププログラムはPostgreSQL ODBC Driverのインストー

ルと同時に英語版のODBCドライバマネージャもインストールします。ご注意ください。

オリジナル（英語版）のインストール途中でODBCドライバマネージャのインストールに関する選択肢が表示される場合があります。選択に迷った場合には、むやみに英語版にされないためにも次の優先順位で選ぶのが安全です（それでも英語版にされる場合があります）。

Install Driver Manager (with version checking)

Install Driver Manager

インストール先のシステムにすでにPostODBCがインストールされている場合、オリジナル（英語版）のセットアッププログラムはこれを削除（中途半端なことにドライバ本体のDLLファイルのみを削除）してしまい、以後PostODBCが正常に動作しなくなります。これを避けるためには、オリジナル（英語版）のインストールの前にPostODBCのドライバ本体のDLLファイル（WindowsのシステムディレクトリのPODBC32.DLL）をバックアップし、インストール完了後に復元してください。

導入が終わると、32bit ODBCコンソール（コントロールパネルから・・・）の「ドライバ」タブのなかに、postgreSQLがあることが確認できます。

<PgAccessのインストール>

TCL/Tkのダウンロード（日本語版）

<ftp://ftp.sra.co.jp/pub/lang/tcl/jp/tcltk805.jp17.zip>

PCACCESSのダウンロード

PgAccess は、[GBorg](#) の [The pgaccess Project -- PgAccess](#) に移り、PostgreSQL のリリースに含まれなくなりました。今後、PgAccess は、PostgreSQL アプリケーションとして、PostgreSQL のリリースとは独立に、[GBorg](#) で開発が続けられます。

ただ、ダウンロードして、解凍し、setup.exeを実行するだけです。

## 9. APPENDIX. その他

<LESS>

デフォルト環境だと日本語表示できません。 [less290i.tgz](#)をダウンロードし、  
/tmp  
に移動します。

```
% tar xvfz less290i.tgz % cp less.exe /usr/local/bin  
で、lessで日本語表示可能です。
```

## WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

以下に、nkfをインストールする方法もかいておきます。(rpmから)

<秀丸などのエディター>

秀丸をCygwin上で使うには、以下のファイルを作ればOK!

```
-- /usr/local/bin/hidemaru.bat --
```

```
"c:\Program Files\Hidemaru\hidemaru.exe" %1 %2 %3 %4
```

---

```
-- /etc/profile --
```

```
...  
alias hidemaru="hidemaru.bat"
```

```
...
```

秀丸以外の場合、hidemaruを適宜変えてやってみてください。

<INVOKER>

<http://www.idetix.com/files/invkr153.zip>

(Windows2000の「サービス」で、自動立ち上げができるようになります。)

<RPM>

(RPMとは、RedHat Package Managerの略で、一般的に利用されるソフトを簡単にインストールし、さらにインストールの履歴をデータベースで管理するものです。なお、RPMに関する詳細については、<http://www.rpm.org/>を参照して下さい。あるいは、市販の本かな・・・)

[Project HeavyMoon](#)から、最新のcygwin用RPMを入手しましょう。[ここ](#)が、rpmのtar+bz2形式最新圧縮ファイルです。ダウンロードして、Cygwinのルートディレクトリにおいてみましょう。

解凍の仕方は、

```
bash-2.03% bzip2 rpm-3.0.6.bin.tar.bz2 | tar xvf - -C /
```

として、解凍します。(rpm-3.0.6.bin.tar.bz2は、ダウンロードしたファイル名を入れます。適当に書き換えてください。)

Windows98等をご利用の方は、Project HeavyMoonにある、patchをあててください。また、9x系でコンパイルがすでにされているモノはみあたらないので、SPECファイルをダウンロードし、ご自身でソースを持ってきてコンパイルしてしまってください

い。

私は、ここから、nkfやinfo、grep、zip、lhaなどを入手し、インストールしましたが、うまく稼動しています。

```
bash-2.03% rpm -ihv *.rpm
```

で、

```
bash-2.03% #####
```

と表示されれば、うまくインストールされたことになります。

インストールされたパッケージ一覧は

```
bash-2.03% rpm -aq
```

で表示されます。

・・・では、ご健闘をお祈りいたします。ちなみに、[「WINDOWS最適環境構築」](#)に紹介する、FileVisor4とメモ帳トラッパー (or DUMMYPAD) +WZ\_EDITOR環境だと、ファイルの中身の漢字コードを意識しなくてすむばかりか、tar.Zファイルやbz2ファイルなど、殆どの圧縮形式のファイルの解凍が楽にできます。(FileVisor5の方が断然良いです。圧縮ファイルをツリー表示できます)

近々、PHPでのpostgreSQL接続 (Windows+Cygwin) の方法についてお教えいたします。(いやー、これが出来たおかげで、開発環境がかなりよくなりました・・・)

## 10. APPENDIX2. もろもろ

実は、PHPのWindows版のほうについては、色々と悩んでいたことがあって、それが解決するまでは、PHP+Windowsのことをかけないな、と思っていたのですが、解決いたしましたして・・・。もうすぐ、PHP+PostgreSQL(Windows + Cygwin)のことを書きますね。以下に、「悩んでいたこと・・・」をPHP-JPメーリングリストにて投稿し、自己解決したものを記します。

Terra-Internationalの北島です。  
自己レスですが・・・。  
以下の件なんですが、解決しました。  
ただ、解決した、といただけでは、情報としてなにも意味がないので、説明を。  
<前提>  
1 PHP4.04 (PHP 4.01以上) for Win は、ISAPI (php4isapi.dll) を使う場合 (つまり、CGI版ではない場合)、inetinfo.exeのcpu使用率が100%になったり、といったバグが、海外では報告されています。  
(PHP 4.02以上だと解決しているというのもあったような気はしますが、そうではなかったみたいです。まあ、PHP4.04でも、「たまには」安定することがありましたけど。)

## WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

ちなみに、cpu利用率が100%になった場合は、タスクマネージャーでinetinfo.exeを選択し、(もしVC++デバッガーがあれば)デバッグをする、にして、デバッガーを閉じると、元に戻ります。一応、はまった人向け情報。(私はこれで何回も助けられました・・・)

2 IISは(バグなのかわざとなのかは釈然としませんが)、CGIを起動する場合(例えば、PHPであれば、php.exe)、HTTPヘッダーが

"302 Object Moved"をだすものに関しては、強制的に、ヘッダー情報を書き換えてしまいます。

つまり、Set-Cookieヘッダーも無視

されてしまいます。

#参照:

#<http://support.microsoft.com/support/kb/articles/q176/1/13.asp>

これがどういう結果をPHPにもたらすか、というと、以下のメールのとおり、SetCookieしてからHeader("Location: ...")で

リダイレクトをかけると、失敗します。

具体的には、

```
| HTTP/1.1 302 Object Moved
| Location: (URL)
| Server: Microsoft-IIS/5.0
| Content-Type: text/html
| Content-Length: 165
|
| <head><title>Document Moved</title></head>
| <body><h1>Object Moved</h1>This document may be found <a
| HREF="(URL)">here</a></body>
```

となります。

#本来は、Set-Cookie: \*\*\*\*\* 行が入るというのに・・・。

<解決方法>

CGI版を使い(つまり、拡張子のマッピングには、php.exeを使い)、このような記述をします。

```
---- (filename:) nph-hoge.php -----
<?
$time = mktime()+600; // Cookieの有効時間が10分と想定
$date = date("l, d-M-y H:i:s", ($time));
$key1 = "cookienam1"; // Cookieのキー名1
$val1 = rawurlencode("対応する値1"); // 対応する値1
$key2 = "cookienam2"; // Cookieのキー名2
$val2 = rawurlencode("対応する値2"); // 対応する値2
$protocol = "http"; // プロトコル
$location = "redirect_to.html"; // リダイレクトしたいファイル名
$cookie = "Set-Cookie: ${key1}=${val1}; path=/;
expires=${date} GMT;\r\n";
$cookie .= "Set-Cookie: ${key2}=${val2}; path=/;
expires=${date} GMT;\r\n";
$header = $protocol . "://" . getenv("HTTP_HOST") . "${location}\r\n";
if ($cookie){ $header .= $cookie . "\r\n";}
Header("HTTP/1.0 302 Redirect\r\nLocation: $header");
```

```
Exit;  
?>
```

尚、注意点は、PHPファイルの名称は、必ず「nph-」をつける、ということです。その点を忘れてしまうと、はまります。

(nph = non-parsed header)

この方法は、(PHP-MLには関係ありませんが) Perl (CGI版) でも有効でしょう。

では、失礼します。

P.S. どなたか、Zend Encoder Unlimitedを使ってらっしゃる方、いらっしゃいますか? Windows 2000 + IIS 5.0 でも動きました。Zend.comには、Win版Apacheしか記述されていませんでしたが・  
・。PHPスクリプトをプリコンパイルして、コードを見えなくする手法は、画期的ですね。ちょっと高いですが・・。。。

--

On Tue, 30 Jan 2001 22:43:36 +0900

Tetsuya Kitahata wrote:

>

> Terra-Internationalの北島と申します。

>

> Windows版のPHP4を導入しているのですが(Win2000 + IIS 5.0)、

> php4isapi.dll ではなく php.exeを、IIS 5.0において.php 拡張

> 子に関連付けたところ、SetCookie関数が通らなくなりました。

>

> 具体的な例としては、

>

> ----hoge hoge.php-----

> <?

>

> .....  
> SetCookie("key", \$value);

> Header("Location: \*\*\*\*\*.php");

>

> .....  
> ?>

>

>

> です。つまり、Location: をHTTPヘッダーに挿入し、リダイレクション

> をかける場合に措いて、です。HTTPヘッダーから、Cookie行の情報が

> すっぱり消えてしまいます。(IISが勝手に書き換えているような気が

> しなくもありませんが)

> この現象は、HTTPのログをみて確認済みです。

>

> そして、IIS の設定に措いて、拡張子関連付けに、DLLを使うと、これは

> 解決します。SetCookie後、Header("Location: ....")しても、正常どおり、

> Cookieは焼かれます。

> #ただ、DLL使うと、なぜか、inetinfo.exeがCPU使用率をものすごく

> #食ってしまったたり、Access Violation エラーが起こることが

> #あるので、これまた困っていますが。

## WindowsでUNIX環境を!! -- Cygwin on Windows, etc.

>  
> --  
>  
> 同じような現象がおこるかた、いらっしゃいますでしょうか？また、  
> 解決策はありますか？  
> 多分、仕様上のバグのようにも思えるのですが、すっきりさせたいと  
> 思っていて、この件につきご存知の方がいらっしゃれば、お教え  
> いただければ幸いです。