



第6章

開発支援はオープン・ソースで

岸 哲夫

この章ではGCCで開発するための環境整備をオープン・ソースのフリー・ソフトウェアを利用して行う方法を説明する。高価なソフトウェアを買って、サポートに頼るのも一つの道だが、フリー・ソフトウェアを使って、自分でネット上の情報を探して解決するほうが勉強になることだろう。(筆者)

開発に役立つツール zsh

zsh とは

zsh はシェル^{注1}の一種です。開発時に大いに役に立つ便利なソフトウェアです。

Linux では大きく分けて csh 型(csh , tcsh)と sh 型(sh , ksh , bash , zsh)という2種類のシェルがあります。Cygwin では、bash を使っています。zsh は後者の分類に入られます。

zsh という命名は「最後のシェル」ということで名付けられたものです。機能としては ksh + csh という形になります。sh / csh / tcsh / bash の機能をほぼ網羅しています。

zsh は Paul Falstad 氏、Richard Coleman 氏、Karsten Thy gesen 氏らによって1992年から開発されて、GPL(GNU General Public License)によって配布されています。本稿執筆時(2006年1月)の最新バージョンは4.2.1です。

zsh は ksh + csh の機能を組み入れ、さらに強力なシェル・スクリプト言語として設計されたシェルです。基本的な UNIX、または UNIX 互換シェルの知識があれば使いこなすことができるツールです。ただし、初心者向けとは言えないので注意して使ってください。

公式サイトは以下のとおりです。

<http://zsh.sunsite.dk/>

zsh は非常に有用なツールなので、たいていの Linux にも最初から入っています。当然のことながら、Cygwin にも入っています。

最初からインストールされている場合には設定ファイルが /etc の下に入っているのが自動的にそれが使われます。もちろん、それを変更してもかまいません。しかし、通常はユーザ

ごとの環境としてホーム・ディレクトリの設定ファイルを修正すべきです。

zsh が起動時にコマンドを読み込む初期化ファイルは、以下のように五つあります。

```
$ZDOTDIR/.zshenv
$ZDOTDIR/.zprofile
$ZDOTDIR/.zshrc
$ZDOTDIR/.zlogin
$ZDOTDIR/.zlogout
```

この順番で有効になります。環境変数 ZDOTDIR が設定されていなければ、HOME の値が使われます。

各設定ファイルの役割は次のとおりです。

▶ .zshenv 環境変数を設定する

最初に設定すべきものをここでを行います。uname コマンドで OS 依存の設定や、hostname コマンドを使ってホスト依存の設定をするには、ここに記述するとわかりやすいはずですが。

▶ .zprofile .zshrc より前に読み込まれるということを除いて、zlogin と同様

.zprofile と .zlogin のどちらかに .zshrc の前に設定しておくべき内容を記述するべきです。両方に記述するのは混乱の元になります。

▶ .zshrc リソースの設定を行う

zsh を直接使っている時だけ使う設定を行っておきます。

たとえば setopt, alias, bindkey などです。

▶ .zlogin ログイン時のみに実行したいコマンドを記述しておく。ログイン・シェルの場合のみ読み込まれる

たとえば ssh-agent を起動しておきたいなら、ここに記述します。

* * * *

zsh は多機能なので、基本機能だけ紹介します。Linux ユーザならば bash を意識せずに使っていると思いますが、そのまま

注1：シェルとはコマンド操作を受け付けて、与えられた指示を OS に伝えるソフトウェアのこと。

図1
コマンドの置換例

```
linux% ls -l $(echo /etc/samba)
合計 28
-rw-r--r--  1 root    root          42 11月 29 10:38 MACHINE.SID
-rw-r--r--  1 root    root          20  4月 11  2002 lmhosts
-rw-----  1 root    root        8192 11月 29 11:09 secrets.tdb
-rw-r--r--  1 root    root        2277 11月 29 10:36 smb.conf
-rw-----  1 root    root         101 11月 29 11:05 smbpasswd
-rw-r--r--  1 root    root          97  4月 11  2002 smbusers
linux%
```

図2
\$変数の内容でコマンド
を実行の例

```
linux% echo $TTY
/dev/pts/0
linux% ls -l $TTY
crw--w----  1 kishi   tty          136,   0 1月 29 13:31 /dev/pts/0
linux%
```

図3
/etc/sambaを testdir と
いう変数にセットして処
理する例

```
linux% setenv testdir /etc/samba
linux% ls $testdir
MACHINE.SID lmhosts secrets.tdb smb.conf smbpasswd smbusers
linux%
```

```
linux% ls -l
合計 0
---x--x--x  1 kishi   kishi        0 1月 29 14:02 test
linux% chmod 755 *(x)
linux% ls -l
合計 0
-rwxr-xr-x  1 kishi   kishi        0 1月 29 14:02 test
linux%
```

図4 ディレクトリ内の実行形式データの属性をすべて変更する例

```
linux% pwd
/dev
linux% ls sda<2-5>
sda2 sda3 sda4 sda5
linux% ls sda<10->
sda10 sda11 sda12 sda13 sda14 sda15
linux% ls sda<-10>
sda1 sda10 sda2 sda3 sda4 sda5 sda6 sda7 sda8 sda9
linux%
```

図5 ファイル名に数値が含まれている場合の例

図6
数字を含むファイル名を
数字に従って並べ替える例

```
linux% echo sda<->
sda1 sda10 sda11 sda12 sda13 sda14 sda15 sda2 sda3 sda4 sda5 sda6 sda7 sda8 sda9
linux%
```

zsh に変えても大きな問題はないはずです。それは、bash の機能はすべて含まれているからです。また、csh を使いこなしているベテランの人でも zsh に満足すると思います。作り込んだ .tcshrc から .zshrc に変換するのも、また楽しいものです。

プロセス名・コマンド名の置換

たとえば、図1のようなコマンド名の置換が可能です。

また、図2のように \$ 変数の内容でコマンドを実行することが可能です。

よく使うディレクトリに名前を付けて入力を簡略化することも可能です。図3の場合は、/etc/samba を testdir という変数にセットして処理しています。このように短い名前ではありがたみがありませんが、長い名前のおときはとても便利です。

プログラミングの際に、このようなことも可能です。

```
linux% vi =(sed s/return/return2/g exit.c)
```

上のコマンドだけで exit.c の中に記述された 'return' を 'return2' に変換して vi で編集できます。

which コマンドで実体を検索したいのならば、

```
echo =ls
```

とコマンドを打つと /bin/ls と結果が戻ります。

ファイル選択の際の拡張機能

ファイルを選択する際に、細かいところで便利に動作させることができます。

例としてディレクトリ内の実行形式データの属性をすべて変更する例をあげてみます。図4のように簡単にできてしまいます。

また、図5のようなことも可能です。図5はファイル名に数値が含まれている場合に便利に使えます。

NUMERIC_GLOB_SORT オプションを使うと数字を含むファイル名を数字に従って並べ替えることができます(図6)。

なお '**/*' という文字列でサブディレクトリを再帰的に取り扱うことが可能です。図7の例なら ls -R でもできます。

もう一つ上の階層に移動して、cソースの一覧を表示してみ