



よしだともこの Linux 事始めの書

第11回 UNIXネタにワイ・ワイ・ワイ

— その1 シェルスクリプトを書こう！

お久しぶりです。My Gardenでは、いま、いろんな色や大きさの朝顔が毎朝30~40も咲いていて、とっても幸せな気分です。

京都ノートルダム女子大学 よしだともこ

<http://www.notredame.ac.jp/tyoshida>

✂ My Happy ATMスイッチLife ”

知り合いとか初対面の人とかと話をしているうちに、共通の知り合いがいたり、以前、すでにどこかで出会っていたことが判明して「世の中、狭い……」となることって多くないですか？

私の場合、最近、特にそういうことが多いのです。しかも、普通に京都市営地下鉄に乗っていて、昔の知り合いにひょっこり出会うこともとっても多い……。

理由はきっと「歳をとったから なのでしょう……(笑)。つまり、生きてきた時間が長いから、それに比例して、世界の人口に占める自分の知っている人の割合が高くなったということなのでしょう。それと、私が京都という狭い土地にどっしり腰を下ろしていることや、取材とかイベント参加という名目でフラフラ歩くことが多いのも影響しているんだらうなあ。

不思議なのは、私がたまたま口にした質問で、共通の知り合いがいたことや、実は古い知り合いだったことが判明することが多いことです。あとで「なぜあのとき、あの質問をしたのか……」を考えても理由はないのですが、「なぜかふとその質問をしてしまう、そして判明する」というストーリー展開なのです。これって「女性の勘」というヤツなのでしょうが……。あるいは、私が暇さえあれば質問をしているために、数打ちゃ当たる式に当たってるだけなのでしょうが。

先日、ある人が「よしださんは、ATMスイッチみたい」とか言っていました。理由は、このスイッチにつながると知り合いが増えるからだそうです。ふふ、それだけだと思えば甘い。私と知り合うと、普通なら単なる会話で終わることが、いつ雑誌で紹介されるか分からない危険にさらされますよ。さっそく「ATMスイッチみたい」を、ここに書いてるしね:-)。

✂ UNIXや教育を熱く語る……

突然ですが、ここでちょっと宣伝です。秋に、京都ノートルダム女子大学の公開講座として、「インターネット時代の家庭や学校の役割および教育を考える(仮題)」という講義を、学外の方を対象に、参加費無料で実施することになりました。せんえつながら、私がコーディネーターおよび講師を務めさせていただきます。

2回コースで、どちらか、あるいは両方を受講していただくこととなります。1回目の10月28日土曜日の午後には実施する「家庭、学校のインターネットの環境が意味するものとは」では、家庭から使うインターネットの意味や役割、そして小中学校での教育に活用されるインターネットの現状や将来像を扱います。

そして、11月11日土曜日の午後には実施する2回目では、「UNIXおよびLinuxの歴史と役割、そして将来展望」をテーマとします。大学のネットワーク構築におけるUNIXの役割、英語教育におけるインターネットおよびUNIXの役割、ノートルダムでのネットワークリテラシー教育の実態などについて、関係者の方々からの事例報告や解説を中心に、最初から最後までUNIXをネタにした講義にしたいと思っています。

両日とも、講義の休憩時間を利用して、大学の教育システムおよびサーバ環境の見学ツアーを実施する予定です。詳しくは、京都ノートルダム女子大学のエクステンションセンターのページ(記事末RESOURCE 1 参照)あるいは、<http://www.notredame.ac.jp/~tyoshida/extension.html>をご覧ください。より詳しく紹介しておきましたので……。

夏の京都は暑くて大変ですが(盆地なので、蒸し暑い……)。 “秋の京都”はなかなかいいですよ。ちなみに、京都ノートル

ダム女子大学(通称、ダム女)というのは、京都駅から京都市営地下鉄で北に15分程度の「松ヶ崎駅」から徒歩5分程度の場所にありますので、交通の便もわりと良いです。

☪ 火曜日はUNIXの日

実は……。春から夏までの数カ月の私 毎日、職場(ダム女のコンピューターセンター)に張り付く運命になった私は、この連載を休載しないといけないは、メールを満足に読み書きする時間はないは、ほかにいろいろとやるべきことを保留にして迷惑をかけてるは、という状況の中だというのに、密かに職場でUNIXファンを増やす活動に励んでおりました。

具体的には、私がコーディネーターという立場となり、コンピューターセンター関係者や知り合いの方々の多大な協力を得て(おもいきり巻き込んで)、毎週火曜日の夕方、「UNIX day」と称する勉強会のようなものを実施したところ、学生と教職員の方が集まってくれました。

「Linux day」ではなく「UNIX day」なのは、Linuxに特化した内容を扱っているのではなく、あえて、UNIX全般に通用する内容を勉強することを目標としているからです。実際、使っているOSはSolarisですし……。

Linuxが流行り出してからというもの、UNIXと同意語でLinuxが使われることが多いじゃないですか。私自身「ここはUNIXと言うべきところだけど、Linuxと言った方が、この人には通じやすいだろう」と、わざとLinuxという言葉を使ったこともあったし、書籍や雑誌記事でも、UNIX一般に通じる話を、まるでLinuxの話のように書いてしまったことがあることは認めます。でも最近、ちょっと目が覚めたんですね。つまり、

Linuxがないころから、UNIXに付き合ってきた者は、これはUNIX全般の話、これはLinuxに特化した話ということが体で分かるからいいけど、そうでない多くの人は、それらをごっちゃにしていまいがちだ……。たまたまLinuxが流行しているからって、それはマズイ……。

みたいなことをマジで考えるようになりました。ということで、今後は私も気を付けて使おうと思いますので、みなさんも、一度、気にしてみてくださいね。

さて。本題のUNIX dayですが、5月の連休明けのスタート時には、次のような案内を作って、学内で広報しました。

このコンピューターセンター研究室には、サーバ類のほかに、学生が使える端末が25台ほどあって、そのうちの10台は古いX端末、残りはすべて、PC UNIXとWindowsのデュアルブートマシンです。

毎週火曜日「UNIX day」のお知らせ

京都ノートルダム女子大学コンピューターセンター研究室は、毎週火曜日の午後3時から6時頃まで「UNIX day」とします。基本パターンは、こんな感じです。

15:00 ~ 16:30

よしだがこの部屋に待機し、学生からのUNIXに関する質問に答えます。

16:40 ~ 18:00

主にUNIXに関して、毎回、テーマを決めて講習会を実施します。講師は、システム管理をお願いしている、京都大学の津邑(つむら)さんをはじめとする、豪華講師陣です。以下が、テーマ候補ですが、学生の希望によって追加・変更します。

・ Window Managerを変えてみよう

Window Managerとは、UNIXの画面のデザインのことです。おしゃれなものに変えて、友だちに差をつけてみませんか？

・ Muleの便利な使い方

メールの読み書き、ホームページ作りに利用されているMuleを、もっと便利に使いませんか。

・ UNIXコマンド、シェル入門

UNIXコマンドやシェルの機能を少し知るだけで、ノートルダムでのメールの読み書きやホームページ作りが、もっと快適になりますよ。

・ CGIつきのWebページを作ろう

CGIつきのWebページとは、アンケート収集や掲示板のことです。プログラミング言語を利用して、このようなページを作ってみませんか？

・ Wanderlustを使ってみよう

ノートルダムでは多くの学生がメールの読み書きに、「Muleを開いて、mh-rmailと入力して……」という方法を使っていますが、もっと別の(新しく、おしゃれな)方法があります。そのうちの一つがWanderlustです。

まず1回目のテーマは、UNIXとは直接関係ないのですが、いろいろ背景があって「パソコンを分解してみよう！」になりました。講師として、本誌の連載もされているCobalt Users Groupの安田豊(やすだゆたか)さんに来ていただいたところ、20名近くの学生が集まる大盛況となりました。その翌週からは、この大学にシステム管理のアルバイトに来ていただいている、京大の津邑公暁(つむらともあき)さんが、講師を担当してくださっています。

テーマは、

第2回「Window Managerを変えてみよう！」	5月23日
第3回「UNIXコマンド・シェル入門」	5月30日
第4回「Mule(Emacs)入門」	6月 6日
第5回「X-Faceの画像を作る」	6月13日
第6回「シェルスクリプトを書く」	6月20日
第7回「LaTeX入門」	6月27日

というふうに順調に進んでいます。「まずは派手なことをやろう」、「UNIXコマンドとEmacsがある程度使えないと何も始まらないでしょ」、「もっと楽しいことした〜い」、「もっと実用的なことした〜い」という感じで、こんな順序になりました。

毎回の詳しい報告書を、UNIX dayの案内ページ[2]から読めるようにしていますので、それぞれの詳しい内容、および、その後のテーマや内容に興味を持った方はそちらをご覧ください。

参加者の人数は毎回5~10名ぐらいで、私も学生気分です。

「あれえ〜、ファイルのコピーができひ〜ん。げー、実行するディレクトリを間違っただけやった……。他人のディレクトリにファイルをコピーしようと思っても、できひんで当然やんなあ……。恥ずかし〜」

などと大声で言いながら、楽しんでます。私って基本的に、先生をしているよりも、生徒として(弟子として)人に甘えてる方が得意なタイプなので。学生たちも、授業中とは違うこの自由な雰囲気を楽しんでくれているようなので、これからも津邑さんを中心とした講師陣が、楽しく講師を担当し続けてくださることを切に願っています。

で、今回の記事では、6月20日のUNIX dayの「シェルスクリプトを書く」の内容を中心にお届けします。

✂ ファイル名を一括変換

「UNIXの便利さを実感してもらうためには、人間様が楽できる例を見せることが先決でしょう……。」ということで、ファイル名を一括変換するシェルスクリプトを書くことになりました。

デジカメが勝手につけてくる、桁数の多いファイル名を、シンプルなものに直してしまおうというわけです。例えば、

```
P0001.JPGというファイル名をa01.JPGに変更し、
P0002.JPGというファイル名をa02.JPGに変更し、
P0003.JPGというファイル名をa03.JPGに変更し、
⋮
P0099.JPGというファイル名をa99.JPGに変更したい
```

といった場合は、ファイル名変更の操作を、99回、人間が手動で実行するよりは、シェルのプログラムを書いて実行させた方が早いですよね。その上、99回が1000回になったり、10000回になっても、プログラムならへっちゃら、だからです。

今回使われるスクリプトはこんな内容

この場合に書かれるスクリプトは、こんな感じになります^{*1}。

```
for i in *.JPG (1行目)
do mv $i `echo $i | sed 's/P00/a/'` (2行目)
done (3行目)
```

それぞれの行でやっていることを具体的に説明すると、次のようになります。

・1行目でやっていること……

“i”という変数に“*.JPG”に合致するファイル名(JPGという拡張子で終わるファイルはすべて合致する)をセットして、合致するファイルの数だけ、2行目以下の処理を実行します。

・2行目でやっていること……

“\$i”には、“i”にセットされたファイル名が入っています。後半の“から”の部分には、echoというコマンドと、sedというコマンドの組み合わせで、「\$iの内容であるファイル名のP00の部分を変えてaに置き換えた結果」を得ます。なぜなら、“sed 's/P00/a/'”というのは「P00という部分をaに置き換える」命令をするコマンドだからです。

その後、mvコマンドで、\$iのファイル名を「P00の部分を変えてaに置き換えた結果」に、変更しています。

・3行目でやっていること……

終了を指示しています。

さて、今回の99個のファイル名変更の場合は、このプログラムが、99周まわる(ループすること)になります。1周目から順番に追っていきましょう。

・1周目に実行されること……

最終的に、「P0001.JPG」を「a01.JPG」に変更したいので、最後に、

```
mv P0001.JPG a01.JPG
```

を実行する必要があります。そうすれば、P0001.JPGという名前だったファイル名が、a01.JPGという名前になるからです。そのために、具体的には、次のような内容が実行されます。

1周目の1行目の「for i in *.JPG」では、“\$i”には、P0001.JPGが入っています。次に2行目の後半の部分「echo \$i | sed 's/P00/a/'」が実行され、a01.JPGというファイル名が得られます。

その後、“mv P0001.JPG a01.JPG”が実行されるので、ファイル名は、「P0001.JPG」から「a01.JPG」に変更されます。

・2周目に実行されること……

今度は、「P0002.JPG」を「a02.JPG」に変更するための処理が

*1 実際にはこのスクリプトでは99個までのファイルしか対応できませんが、説明を簡単にするため、あえてこの例にしました。

実行されます。具体的には、2周目の `for i in *.JPG` では、`$i`には、`P0002.JPG`が入ります。そして、“`echo $i | sed 's/P00/a/'`”が実行されて、`a02.JPG`が得られます。その後、“`mv P0002.JPG a02.JPG`”が実行されるので、ファイル名は`a02.JPG`に変更されます。

このように、「ファイル名の先頭のP00を、aに変える」という規則は、毎回の処理に共通しているので、これをプログラム(スクリプト)で記述すれば、人間は楽できるというわけなのです。

部分ごとに実行してみよう

まず`echo`というコマンドは、その後の文字列を表示する「だけ」のコマンドです。実行してみます。

```
$ echo P0001.JPG
P0001.JPG
$
```

これだけ見ると「`echo`コマンドって、なんの意味があるねん」という感じですが、ちゃんという意味があるから、使うわけですね。

次に、“`sed 's/元の文字列/置換後の文字列/'`”は、文字列を置換させるコマンドです。実行してみます。

```
$ sed 's/P00/a/' リターン
P0012345          という入力文字列を入れてリターンすると.....
a12345           という結果が表示されるので、Ctrl+dで終わる。
$
$ sed 's/P00/a/' リターン
P0001.JPG        という入力文字列を入れてリターンすると.....
a01.JPG         という結果が表示されるので、Ctrl+dで終わる。
$
```

最後に、この2つを、| (パイプ)で組み合わせて、`echo`の出力を、`sed`の入力にしたのが、次の実行例です。

```
$ echo P0001.JPG | sed 's/P00/a/'
a1.JPG
$ echo P0001.JPG | sed 's/P00/a/'
a01.JPG
$
```

こんなふうに部分ごとに実行することで、だんだん、最初のスクリプトの内容が理解できてきましたね。

シェルスクリプトの3つのポイント

・1つ目

“`*`”はワイルドカード(Wild Card)と呼ばれ、何にでも合致する。例えば、`*.JPG`なら`.JPG`で終わるファイル名がすべてあてはまる。

・2つ目

“`|`”はパイプ(Pipe)と呼ばれ、|の前のコマンドの出力が、|の後のコマンドの入力になる。

・3つ目

“`‘`”はバッククォート(backquote)と呼ばれ、この記号で囲まれた部分が、まず最初に実行されて、その結果が、シェルの処理に使われる。

実行してみよう!

実際に、`P0001.JPG`、`P0002.JPG`、`P0003.JPG`というファイル名のファイルを作って、実際に、このシェルスクリプトを実行してみます。

`touch`コマンドを使えば空のファイルができます。

```
$ ls                               (最初はファイルは何もない)
$ for i in 1 2 3
> do touch "P000$i.JPG"
> done
$ ls
P0001.JPG P0002.JPG P0003.JPG (ファイルが3つできてる)
```

次に、`rename.sh`という名前のファイルを開いて、で紹介したプログラムを書き込みます。

```
$ mule rename.sh &
```

このファイルに、以下の3行を入力し、保存します。

```
for i in *.JPG
do mv $i `echo $i | sed 's/P00/a/'`
done
```

入力し終わったら、`rename.sh`という名前のファイルとして保存します。

シェルのインタプリタはsh

さて、`sh`というのが、シェルのインタプリタなので、“`sh rename.sh`”と入力すると、シェルスクリプトは実行されます。

```
$ ls
P0001.JPG P0002.JPG P0003.JPG rename.sh
```

```
$ sh rename.sh (シェルを実行)
$ ls
a01.JPG a02.JPG a03.JPG rename.sh
$
```

これで見ごと、コマンド名の変換に成功しました。

ちょっと応用問題

今とは逆に、「a01.JPG」を「P0001.JPG」に、「a02.JPG」を「P0002.JPG」に、「a03.JPG」を「P0003.JPG」にするためには、どんなスクリプトを書けばよいでしょうか。

答えは、次の通りです。

```
for i in *.JPG
do mv $i `echo $i | sed 's/a/P00/'`
done
```

これを、rename2.shという名前前で保存して実行させてみます。

```
$ ls
a01.JPG a02.JPG a03.JPG rename.sh rename2.sh
$ sh rename2.sh
$ ls
P0001.JPG P0002.JPG P0003.JPG rename.sh
rename2.sh
```

コマンド名が変換できたら成功です。

スクリプトをコマンドに

自分が書いたスクリプトをコマンドにしたいときは、先頭の行に、

```
#!/bin/sh
```

という行を新しく追加して、次の4行にします。

```
#!/bin/sh
for i in *.JPG
do mv $i `echo $i | sed 's/P00/a/'`
done
```

次に、このファイルのモードを変更します。

例えば、lsとか、muleは、実行できるコマンドです。しかし、普通のファイルは実行できるようにはなっていません。

そのため、実行させるためにはモードを変更する必要があります。変更する前のファイルのモードは、こんな感じです。

```
$ ls -l
-rw-r--r-- 1 tyoshida 2003 0 Jun 20 17:23 P0001.JPG
-rw-r--r-- 1 tyoshida 2003 0 Jun 20 17:23 P0002.JPG
-rw-r--r-- 1 tyoshida 2003 0 Jun 20 17:23 P0003.JPG
-rw-r--r-- 1 tyoshida 2003 57 Jun 20 17:40 rename.sh
$
$ chmod +x rename.sh (モードを変更する)
$ ls -l
-rw-r--r-- 1 tyoshida 2003 0 Jun 20 17:23 P0001.JPG
-rw-r--r-- 1 tyoshida 2003 0 Jun 20 17:23 P0002.JPG
-rw-r--r-- 1 tyoshida 2003 0 Jun 20 17:23 P0003.JPG
-rwxr-xr-x 1 tyoshida 2003 57 Jun 20 17:40 rename.sh
$
```

こうしておくことで、今後、コマンドとして実行できるようになります。

```
$ ./rename.sh (このディレクトリにあるrename.shを実行)
```

めでたし、めでたし。

最後に難題が.....

シェルスクリプトを書く実習を終えて、めでたし、めでたしという気分のところ、ある学生からこんな質問が出てきました。

「あの～。私がデジカメのファイル名を変更したいときというのは、いつも自宅のWindowsパソコンを使っているときなので、UNIXのシェルスクリプトは使えないと思うんですが、そんな場合はどうすればよいでしょうか.....」

ガーン!!

みなさんなら、どう答えますか？ もし良い案が浮かんだら、tyoshida@notredame.ac.jpまでお寄せください*2。

UNIX dayは今のところ毎週、続いていますので、次回もまた、このネタをお送りできると思います。

ほかにも、私の大学での自分の環境づくりネタとか、zxLinux([3])が動いているシャープのザウルスをゲットした話だとか、えらく([4])の運営サーバ(メーリングリストおよびWebページの管理)を、3月上旬に、それまでお世話に

*2 編集部注：例えばWindowsでは、「Win Rename (<http://www.antenna-a.com/>)」のようなソフトウェアがあります。このように、WindowsやMacintoshでは、何かしたいことがあった場合、「それが出来るアプリケーションを探す」というのが、ユーザーの正しい態度です(もしそれがなければ、基本的にはあきらめるしかありません:-)。しかしUNIXでは、本文で述べられているように、基本的なツールを駆使することで、やりたいことをやりたいように実現できるのが大きな特徴です。

なったレーザーファイブ株式会社から、日本Linux協会〔5〕のシステム部に、お引越させてもらった話だとか、書こうと思いつつも、書いてないままの話はいろいろあるんですけどねえ……。

おまけ：Rubyのมาสコット誕生

Open Source Toys Project〔6〕に、6月下旬、素敵な仲間が加わりました。Rubyのมาสコット、羊の「るうちゃん」です（写真1）。この羊のモデルになっているのは、アスキー発行の書籍「オブジェクト指向スクリプト言語Ruby〔7〕」中の羊です。作成者は、しばたかおさんで、命名者は、Rubyの作者のまつもとゆきひろさんです。「るうちゃん」は「ルビーの首輪」をしているのが特徴です。

しばたかおさんは、昨年、Linux Westのイベント会場で、私たちがお渡しした「ハッスル君キット」で、ペンギンのぬいぐるみを作ってくださって以来、Open Source Toys Project（以下、OSTP）に、多大な協力をして下さっている方の1人です。

OSTPの「ハッスル君」、「なまず君」、「ともこちゃん」そして「るうちゃん」の型紙や作り方のソース一式をまとめて、提供してくださっているのも、しばたさんです。なお、今回、本誌の付録CD-ROMに展開して収録しています*3。

OSTは、5月下旬、OSTL（Open Source Toys License）をリリースして、BSD magazine No.4〔8〕の「プロジェクト短信（執筆者：片桐麻里子さん）」に、3ページにわたって掲載しました。このライセンスは、非常に柔軟性があり、かつ人の良心を信じたものになっています〔9〕。

ちなみにこのOSTP、田宮まやさんが、Linux Weekly News〔10〕に、英語で紹介して下さったおかげで、海外（オーストラリア）からのプロジェクトへの協力依頼もありまして、急ぎよ、英語で情報を交換するメーリングリストを立ち上げることになりました。興味のある方は、hs1@tomo.gr.jp（HSL本、読者の会）まで。まずは日本語で。



写真1 るうちゃん

Ru-u chan' was originally designed by SHIBATA Takao (OST Project). See also... <http://ost.w3.to/>

しばたさんも、英語（と日本語）で、ページを作られました〔11〕。

ここ数カ月、私がこのプロジェクトのことが何もできない間に、どんどんと発展して、インターナショナルにもなって、スタートさせた本人としては非常に光栄です。

6月10日、11日に名古屋大学で開催された「FreeUNIXの集い」への訪問に続いて、9月9日に長野県塩尻市で開かれる地域コミュニティのイベントにおじゃまします。詳しくは、<http://linux.ncs.gr.jp/>を参照のうえお越しください。

R E S O U R C E

- [1] 京都ノートルダム女子大学のエクステンションセンターのページ
<http://www.notredame.ac.jp/extension/>
- [2] UNIX dayの案内
<http://www.notredame.ac.jp/~tyoshida/unix/>
- [3] zxLinux
<http://www.zxlinux.org/>
- [4] えるらく (Ladies' Linux Users Group)
<http://llug.linux.or.jp/>
- [5] 日本Linux協会 (JLA)
<http://jla.linux.or.jp/>
- [6] Open Source Toys Project
<http://www.tomo.gr.jp/ost/>
- [7] 「オブジェクト指向スクリプト言語Ruby」
まつもとゆきひろ、石塚圭樹著 / 株式会社アスキー発行 / ISBN4-7561-3254-5 / 1999年11月
- [8] ASCIIムック BSD magazine 2000 No.4
株式会社アスキー発行 / ISBN4-7561-3458-0 / 2000年3月
- [9] Open Source Toys Licenseについて
<http://www.tomo.gr.jp/ost/OSTL.ja>
<http://www.tomo.gr.jp/ost/OSTL.en>
- [10] Linux Weekly News掲載のOSTP紹介記事(英語)
<http://lwn.net/2000/features/ohpa-ost/>
- [11] Unofficial Open Source Toys Project Page
<http://ost.w3.to/>

*3 ソース一式をRPM形式でまとめたものは、<ftp://ftp.ring.gr.jp/pub/linux/Vine/VinePlus/2.0/RPMS/noarch/open-source-toys-1.01-0v13.noarch.rpm>から、tar.gz形式のものは<http://www.tomo.gr.jp/ost/>から入手できます。