

# IT 特別講座

## 初心者のためのLinuxベーシック 予習篇

---

(自宅でお気軽Linux)

LA-Linux 専任講師 矢越昭仁

2009/03/29

Windows PC で Linux が利用できる環境の紹介と、実際に TurboLinux を使って環境を立ち上げます。その上で vi を使ってシェルスクリプトを作成・演習を行います。

## 目次

Linuxという OS.....	2
自宅で Linuxを楽しむには .....	5
インストールする方法.....	6
デュアルブートによる方法.....	6
Live CD/USB を使う方法.....	7
仮想化ソフトを使う方法 .....	7
クラウド・コンピューティングを使う方法 .....	8
エミュレータを使う方法 .....	8
ちょっとかじってみる .....	9
BIOSの変更.....	9
Linuxの起動.....	9
まずは基本コマンドの確認.....	10
自分の環境を調べる .....	10
ファイルを調べる.....	11
何をするにも必要な vi.....	12
vi の特徴.....	12
起動と終了.....	12
シェルスクリプトを作る .....	13
シェルスクリプトを動かす.....	14
スクリプトの改良（ 1 ） .....	14
スクリプトの改良（ 2 ） .....	16
付録 1：主な vi コマンド一覧.....	19
付録 2：Cygwin.....	20
付録 3：Vim.....	24

## Linux という OS

Linux は 1960 年代に開発された UNIX を祖先とし、1990 年末に当時フィンランド・ヘルシンキ大学生 Linus Torvalds さんが開発し公開した基本ソフト(OS: Operating System)です。祖先である UNIX は長い歴史の中で、紆余曲折があり利用形態や著作権、IT 業界の政治的駆け引きなど数多くの問題を抱えていました。そのしがらみをリセットしたのが Linux です。

Linux はインターネットの普及とあいまって、世界中の有志により次々に機能が追加され、動作するコンピュータの種類も増え適応できる範囲が飛躍的に拡大しました。これは Linux がその設計図ともいえるソースコードを無償で公開したためだと言われます。Linux や UNIX の周辺では他にも FSF/GNU Project がプログラミング言語を中心としたツール類、

Apache Project が Web サーバーを中心としたサーバー群を無償で公開しています。これらの公開されたソフトウェアを OSS(Open Source Software、ソース = 設計図が公開されたソフトウェア)と呼び、今ではコンピューター・メーカーの枠組みを超えた一大勢力となっています。

## ディストリビューション

Linux そのものは無償で提供されていて、いつでも・どこでもインターネットから入手できますが、そのままでは準備すべきソフトウェアが大量になり使いこなすのが大変です。しかも、それらソフトウェアの設定を個別に行わなくてはならないとすると、適切な組み合わせを選び、動作確認するなど膨大な作業が必要になります。たとえばベーシックコースの実習に使っている Linux には、900 個以上のソフトウェアがインストールされています。

そこで登場したのがディストリビューションという方法です、ディストリビューションは核となる Linux 本体(カーネル)と普段よく使うであろうソフトウェア群を組み合わせ、セット提供しています。

ディストリビューションには、システム管理ツール(特にパッケージ管理ツール:ソフトウェアの構成管理、インストール、アンインストールを行うためのツール)の違いから、大きく Red Hat 系と Debian 系という2つ流派に分かれます。次の「主なディストリビューション」表には、国内外の代表的なディストリビューションを記載しています。表中の「型」に流派をそれぞれ Red Hat 系を R、Debian 系を D で表現しています。

## 主なディストリビューション

名称	開発元	特徴	最新版	型
Red Hat	Red Hat 社	ディストリビューションの老舗かつ最大手が提供する Linux です。企業向けの色合いが強く製品名 Red Hat Enterprise Linux (略して RHEL) は業界標準となっています。 <a href="http://www.jp.redhat.com/">http://www.jp.redhat.com/</a>	RHL5 07/03	R
Fedora	Fedora Project (Red Hat 社が支援)	RHEL の試作(最新技術の取込)と位置づけられ、更新サイクルが早いのが特徴です。Ubuntu に比べライセンスフリーをより厳密に徹底しています。 <a href="http://fedoraproject.org/">http://fedoraproject.org/</a>	Ver. 10 08/11	R
CentOS	CentOS Lance Davis	RHEL の完全なクローンを目指す多機能・高安定性な Linux を目指し開発されています。そのためサーバー構築向け OS としての位置づけが強くなっています。 <a href="http://www.centos.org/">http://www.centos.org/</a>	Ver. 5.2 08/06	R
Turbo Linux	ターボリナックス社	早期に日本で立ち上がった米国発祥ブランドで、現在は日本法人が事業を買収して運営しています。漢字対応やデスクトップ化で人気を博しました。 <a href="http://www.turbolinux.co.jp/">http://www.turbolinux.co.jp/</a>	Client 2008 08/08	R
Miracle Linux	ミラクルリナックス社	Oracle DB が効率的に動作するようチューニングされた RHEL ベースの Linux です。現在は中国(紅旗)・韓国(ハーンソフト)・日本・ベトナム(ベトソフトウェア)の 4 拠点で Asianux として開発されています。 <a href="http://www.miraclelinux.com/">http://www.miraclelinux.com/</a>	Server 3.0 07/09	R
Debian	Debian Project	中立性、柔軟性が高く、完全なフリーソフトウェアを目指しボランティアが開発を進めています。FSF の GNU プロジェクトの一つで対応 CPU の豊富さも特徴です。 <a href="http://www.debian.org/">http://www.debian.org/</a>	Ver. 5.0 09/02	D
Ubuntu	Ubuntu Foundation / (英)Canonical Ltd.	年 2 回とうまめなバージョンアップと使いやすさを主眼に設計され、他言語対応と幾つかの廉価ノート PC で採用され急速に普及しています。 <a href="http://www.ubuntulinux.jp/">http://www.ubuntulinux.jp/</a>	R 8.10 08/10	D
KNOPPIX	Klaus Knopper、 (独)産業技術総合研究所	Debian をベースにした軽量 Linux で、Live CD を採用しています(後述)。日本語化は産業技術総合研究所が行っています。 <a href="http://www.knopper.net">http://www.knopper.net</a> <a href="http://www.rcis.aist.go.jp/project/knoppix/">http://www.rcis.aist.go.jp/project/knoppix/</a>	6.0.1 09/02	D
Slackware	Patrick Volkerding, Slackware team	古くからあるディストリビューションで、非常にコンパクトな作りになっています。また他のディストリビューションと違いパッケージ管理ツールはありません。 <a href="http://www.slackware.com/">http://www.slackware.com/</a>	12.2 08/12	-
openSuSE	S.u.S.E. GmbH (Novell が支援)	ドイツ発祥の Linux で欧州でのシェアが大きく、独自の YaST パッケージ管理機能を搭載しています。安定性について評価が高いのが特徴です。 <a href="http://www.novell.com/ja-jp/linux/">http://www.novell.com/ja-jp/linux/</a>	11.1 08/12	- (R)
MontaVista	MontaVista Software	組込向け(自動車、携帯電話といった装置に搭載される)Linux のリーディングベンダーです。(旧 Hard Hat Linux) <a href="http://www.mvista.com/">http://www.mvista.com/</a>	5.0 07/03	-

## 自宅で Linuxを楽しむには

Windows PC で Linux を動かすには様々な方法があります。それらを簡単に表にまとめると以下ようになります。

方式	いいところ	不便なところ	お手軽度
インストール	PC の性能を 100%使って、100%の機能を使う事ができます。	Windows は失われます(PC は Linux 専用機となります)。	×
デュアルブート	「インストール」と同様ですが、さらに起動時に切替えて Windows も動作させることができます。	設定が複雑。場合によっては Windows の再インストールを伴います。	××
Live CD/USB	とにかく手軽に動作して、PC に影響が全く出ません。	Linux への設定など高度な設定はできず、ファイルの保存もできません。	
仮想化	Windows からプログラムの一つとして Linux を呼び出すことができます。また複数の Linux や他の OS をインストールする事ができます。	仮想化ソフトによっては Windows の再インストール必要となります。パフォーマンスやネットワークに若干の制約が生じます。	
クラウド・コンピューティング	インターネットを経由して利用するため、インストールする必要はありません。一般的に従量制の料金体系です。	終了する時には次回に設定した内容が反映できるよう、システム環境を保存する必要があります。	
エミュレータ	Windows の普通のソフトとしてインストールでき手軽に利用できます。ファイルの共有も簡単です。	OS 機能(特にシステム設定やハードウェア依存の機能)に大幅な制限があります。	

それでは、各方法について、その概要を解説してゆきます。

## インストールする方法



最近は PC を購入しても、自分でバージョンアップや再インストールをする人は少ない様です。最近ではアップグレードキットも量販店でめっきり見なくなったので、アップグレードせずに PC ごと買い替え、買い増しをしている人が増えたのでしょう。

「インストールする」方法では、今ある Windows PC を潰して Linux 専用機にする方法です。Linux は一般的に Windows よりもパフォーマンスが良いので、使わなくなった古い PC を再利用したい人には良い選択だと思われます。

一昔前までは難しい英語の質問に答えるメニュー形式のインストールでしたが、最近ではどこのディストリビューション(かなりの長きにわたり不親切なインストーラーだと酷評されていた Debian さえ)でも GUI を使った多言語対応のインストールが可能となっています。

インストールするには Linux のキットが入った CD または DVD を用意し、PC の BIOS を変更して CD/DVD から起動できるようにしておきます。Linux のキットをトレーに差し込んで再起動するだけでインストーラーが立ち上がります。質問が分からない場合は、省略値(推奨値)を採用すれば大抵簡単にインストールできるようになっています。

## デュアルブートによる方法

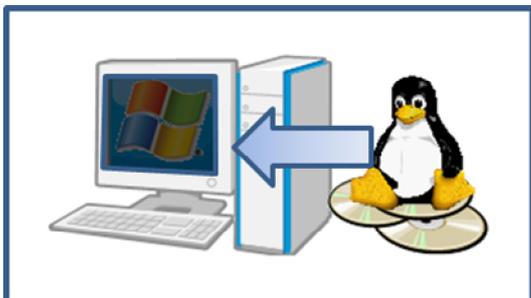


こちらも Linux をインストールしますが、PC 電源投入時に一旦ブートローダーと呼ばれるプログラムが立ち上がるようにしておきます。ブートローダーに指示する事で Windows と Linux のどちらを起動するか選択できるようになります。ディスクの空き容量や Windows のバージョン、利用するブートローダーにもよ

りますが、多くの場合 Windows を削除しブートローダーをインストールする必要があります。

つまり PC を更地にしてからブートローダーをインストールし、さらに Windows と Linux をインストールするという手間が発生します。また既に使っている Windows の環境を引越す手間もかかります。

## Live CD/USB を使う方法



インストール時と同様に BIOS を変更して、CD/DVD や USB から起動できるようにしておきます。インストールと違いライブ形式と呼ばれる Linux が入った CD/DVD や USB メモリーを準備し再起動すると、インストールはせずに小さな Linux が立ち上がります。例えば、Windows は停止した状態で Linux を利用する

形になります。Live CD/DVD の場合、システム変更ができません (CD/DVD に直接、更新・追記ができないため) し、ファイルアクセスのスピードがインストール方式に比べ格段に遅くなるという欠点があります。

今回、の特別講座では Turbo Linux Client 2008 の Live CD を使って解説します。  
(Live CD は無償ですので、そのままお持ち帰れます)

## 仮想化ソフトを使う方法



仮想化ソフトウェアをインストールし、その仮想化ソフトに Linux をインストールする方法です。仮想化ソフトは普段使っている PC の中に、疑似的な PC をもうひとつ構築する技術で、シトリックス社の Xen、ヴイエムウェア社の VMware、マイクロソフト社の Virtual PC などがあります。

仮想化ソフトは大きくホスト OS 型とハイパーバイザ型の 2 種類に分かれます。



ホスト OS 型は、もともと PC に入っている OS (ホスト OS) 上で仮想化ソフトを動作させ、さらに、その上にゲスト OS と呼ばれる OS をインストールします。VMware や Virtual PC はこのタイプになります。

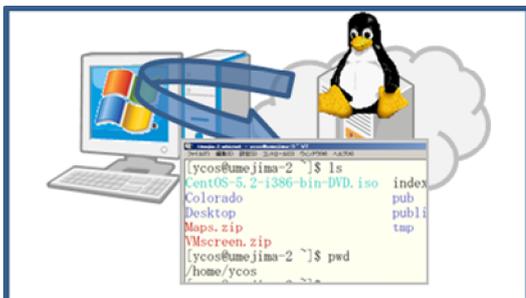
ハイパーバイザ型は仮想化ソフトウェアを OS としてインストールし、その上にゲスト OS をインストールして動作させます。多くのハイパーバイザーは Linux をベースに開発されているため、Windows PC の場合はいったん Windows を潰してハイパーバイザをインストールする必要があります。Xen や KVM はこのタイプになります。

一つの仮想化ソフト上に複数のゲスト OS を配置することもできるので、1 台のマシンで 3 台分の仕事をこなすといったことも可能です。

最近の IT 業界では、この仮想化ソフトを利用して沢山あったサーバーを 1 台にまとめて管

理したり、作業の負荷に応じてコンピュータ・リソース (CPU、HDD など) を動的に配置するとう動きが一般化しつつあります。

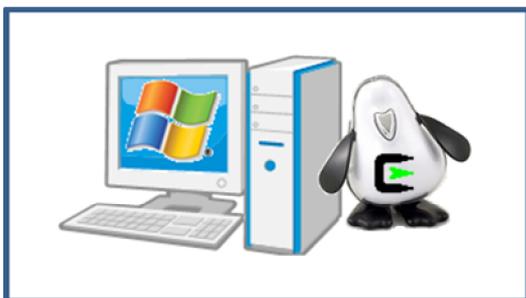
#### クラウド・コンピューティングを使う方法



インターネットを介して、仮想化されたコンピュータを利用する方法で、セールスフォース・ドットコム、Amazon EC2、GoGrid、FlexiScale などがあります。利用するにはインターネットに接続できる環境があればよいだけなので、インストールといった手間は発生しませんし、自分自身の PC に何か変更を加える事もありませ

ん。EC2 など多くのクラウド・コンピューティングは従量課金ですが、非常に安価な価格設定になっている事が多く、利用できるシステム規模も選ぶことができます (EC2 では 1 時間 10 円程度です)。OS やハードディスクを仮想化しインターネット経由で利用することを特に HaaS (Hardware as a Service)、その上でサーバを構築した PaaS (Platform as a Service)、さらにその上に会計ソフトなどを載せてアプリケーションを提供するものを SaaS (Software as as Service) と呼びます。IT 業界では 2008 年頃から爆発的なブームになっています。

#### エミュレータを使う方法



Linux を真似た Windows ソフトウェアを普通にインストールして利用する方法です。何かを真似たソフトウェアの事をエミュレータ (模倣) とよび、たとえば携帯電話をマネたり、ゲーム機をマネしたソフトウェアを使って PC 上でプログラム開発や動作確認を行っています。Linux のエミュレータで有名なのは Cygwin

で、インターネット経由でいろいろな追加ソフトウェアをダウンロードする事ができます。ただしエミュレータはあくまでも、模倣に過ぎないので本物の Linux とは動作が若干異なり、システム設定などはほとんどできません。ただし Linux ベーシック・コース (前半) 程度の内容であれば自習環境として利用できます。

ちょっとかじってみる

まずは最も手軽な方法として、Live CD を使ってみましょう。今回利用するのは日本製の TurboLinux Client 2008 です。操作感が Windows に近く親しみやすい製品でしょう。Linux の Live CD には他にも、Debian Live、KNOPPIX、Fedra、Ubuntu などがあります。またネットワークからダウンロードして、CD や USB に書き込んで利用するタイプのももあります。



多くのディストリビューションは LiveUSB 作成ツールも提供していて、このプログラムをダウンロードして使えば、Windows 上で Linux の Live USB を作成することができます。

<https://fedorahosted.org/liveusb-creator/>

## BIOS の変更

BIOS(バイオス、Basic Input/Output System)は、ハードウェアに組み込まれたソフトウェアで、ハードウェアが故障していないか調べる機能と、それらの入出力を司ります。最近ではセキュリティの関係で、ハードディスクの暗号化や電源投入時のパスワード確認なども行うよう拡張されています。

Live CD/USB を利用するには、この BIOS の設定を変更し OS をハードディスクではなく CD/USB から探し出すよう指示する必要があります。OS がどこにあるかを探す順番のことをブートシーケンス(Boot Sequence)と呼び、どこのメーカーの BIOS にも設定項目があります。なお USB からブートできる PC は比較的新しいものに限られます。

BIOS はメーカーや型番に依存しますが、多くの場合は起動時に[F2]キーを押して、呼び出します。

LA では 5 階の PC が USB 起動対応になっています。また[F12]でブート先を指定することができます。

## Linux の起動

Turbolinux Client2008 が起動すると Turbolinux と Simple の 2 つ選択肢が表示されるので、Simple にカーソルを合わせ (矢印キーの上下と[Enter]) 続きます。

5 分ほどして、デスクトップが表示されると、使用許諾確認画面が表示されるので、内容を

確認のうえ [ 同意する(A) ]をクリックします。さらにブラウザーが起動されますが、これは無視してかまいません。

左下の START メニューから、「システム>端末コンソール」を選択し、起動します。

まずは基本コマンドの確認

Linux ではコンピュータを操作するのに、もっぱらコマンドを使います。コマンドとは「コンピュータへの指示・命令」の事で、Windows ではマウスとアイコンやメニューを駆使している操作に代わるものです。マウスを使った操作を GUI(グイ、Graphical User Interface) とよびます。一方 Linux はキーボードから呪文(コマンド)を打ち込んで、一つ一つコンピュータに指示を出します。この方式は GUI に対し CUI(Character User Interface)と呼ばれています。

先の「端末コンソールを」起動すると、\$マークが表示され、キーボードからの入力待ちになります。この記号をプロンプト(入力促進記号、prompt)と呼び、この記号が表示されている間はコマンド入力が可能です。

```
[turbo@localhost]$
```

コマンドは文字列の並びで、空白で区切られます。空白で区切られた塊を左からコマンド、第1引数(ひきすう)、第2引数...と呼びます。引数のうちハイフンで始まるものは、コマンドの動作を修飾するため、特にオプションと呼ばれます。

```
ls -l /tmp
```

この例では、コマンドが ls で、第1引数が l、第2引数が /tmp になります。また第1引数はハイフンで始まっているので、l オプションとも呼びます。

自分の環境を調べる

それでは起動した Linux について、その状況を確認してみましょう。まずは Linux のバージョンと CPU の内容を確認します。バージョン情報は uname コマンドで確認することができます。

```
$ uname -a
```

```
Linux localhost 2.6.24.7-desktop-1.uc2mb #1 SMP Thu Jul 17 17:22:43 EDT 2008  
i686 Intel(R) Core(TM)2 CPU          6400 @ 2.13GHz GNU/Linux
```

ホスト名(コンピュータ自身の名前)、OS のバージョン、カーネルの作成日時、CPU の種類などが表示されます。

つづいて、今このコンピュータの稼働状況(といっても自分自身の事なので新たな発見はないでしょうけれど)を表示します。

```
$ uptime
```

```
23:30:28 up 4 min, 0 users,  load average: 0.40, 0.69, 0.34
```

uptime コマンドは左から、コンピュータが立ち上がった時刻、稼働時間、ユーザ数、過去

1分、5分、15分の平均負荷(仕事ぶり)が表示されています。Linuxではこの様にコンピュータの利用状況を知るための機能が豊富で、サーバー構築には非常に向いていることがわかると思います。

```
$ id
uid=500(turbo) gid=500(turbo) 所属グループ =500(turbo)
```

Turbolinuxではログイン(使用開始)した状況で起動されますが、一般のLinuxでは、ユーザー名とパスワードを入力する必要があります。現在どのユーザーでログインしているかは、idコマンドで確認することができます。

### ファイルを調べる

Linuxではログインすると、自動的に作業領域としてファイル場所が用意されます。これをホームディレクトリと呼び、普通はその場所で作業を行います。作業場所を確認するにはpwd、移動するにはcdコマンドを使います。またファイルの一覧を表示するにはlsコマンドを使います。

```
$ cd
$ pwd
/home/turbo
$ ls
<ファイル一覧>
```

コマンドを入力すると、一瞬止まったようになる事がありますが、これはLive-CDの欠点です。Live-CDはハードディスクに何も書きこまないかわりに、必要最小限の情報をメモリー中に配置します。メモリーに入りきらなかった、一部のコマンドはCDに格納されています。CDは一般的にハードディスクよりもアクセスが低速(40~50分の1)で、その読み取り時間がかかっている間が止まったように見えるのです。

## 何をするにも必要な vi

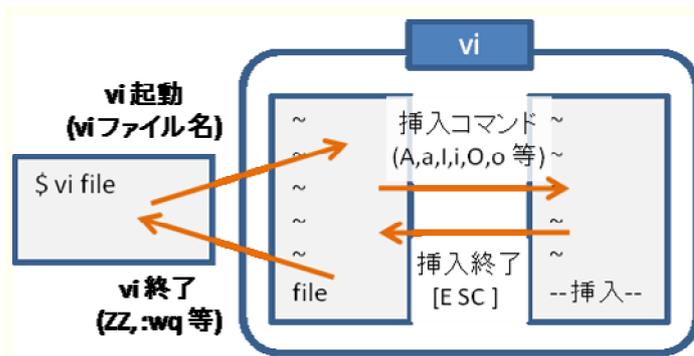
Linux ベーシックコースやマスターコースで必ず必要であり、受講生の皆さんが最も苦手なツールの一つに vi エディタがあります。操作方法が全く普通のエディタと違いますし、マウスなども使えません。しかし、とても重要なので教室でのライブ授業以外に自宅や学校・勤め先でも「触って慣れて頂きたい！」というのが講師の気持ちでもありました。

そこで、今回はこの環境を使って簡単な vi の実習をしてみましょう。

### vi の特徴

vi は普通のエディタとは随分と変わった構造をもっています。このため苦手な人も多いのですが、なれると普通のエディタよりも効率よく編集することが出来るのではないかと著者は思っています。その証拠に Windows でも vim を多用しています。

それでは vi の特徴ですが、なんといってもモードがわかれている点にあります。文字を入力する時と、カーソルの移動や文字の削除・カット&ペーストが明確に分離されています。これを入力モードとコマンドモードと呼んでいます。



さらにコマンドは普通のアルファベットを用いたものが多く、慣れないうちはコマンドを入力しているのか、文字列を入力しているのかを混乱しがちです。

とにかく、「あれ？おかしいな」と思ったときはコマンドモード

へ復帰する[ESC]キーを（何度でも）押してください。

### 起動と終了

vi の起動は、vi に続けてファイル名を指定します。指定したファイルがない場合は新規作成扱い、存在する場合は変更扱いになります。またファイル名を指定しなくても vi は起動することができますが、変更した内容を保存するためには、ファイル名を終了時に指定する必要があります。

```
$ vi /etc/passwd
```

終了するには、コマンドモードで「ZZ」と入力するか、「:wq (上書き終了)」「:q!(保存せず終了)」を入力します。

コロン(:)で始まるコマンドは、ex コマンドと呼ばれ、画面下に表示され[ Enter ]を押すまで実行されません。慣れるまでは ex コマンドの方が確実でしょう。

## シェルスクリプトを作る

Linux は開発当初からグローバル対応（世界規模で利用できる）設計ですので、標準時を変更することも簡単にできます。日時を表示するコマンドは `date` ですが、これを実行する前に `TZ` という変数に地域名を設定すると、その地域の現在日時が表示されます。

```
$ date
2009年  3月 16日 月曜日 16:58:37 JST
```

```
$ export TZ=US/Hawaii
```

```
$ date
2009年  3月 15日 日曜日 21:58:44 HST
```

しかし、一旦 `TZ` を変更するとそのままになるので、`TZ` に `Japan` を指定する必要があります。また、毎回毎回 `TZ=XXX` と指定するのは面倒です。

```
$ TZ=Japan
```

```
$ date
2009年  3月 16日 月曜日 17:02:41 JST
```

```
$ TZ=Australia/Sydney
```

```
$ date
2009年  3月 16日 月曜日 19:02:44 EST
```

こんな時は、キーボードから入力するコマンド群を予めファイルとして用意しておき、まとめて実行する事ができます。このファイルの事をシェルスクリプトと呼びます。

それでは、まず `wdate` という名前シェルスクリプトを作ります。

```
$ vi wdate
```

```
vi エディタが開き、~の羅列が表示される
```

エディタが起動したら、最初に小文字の `a` を押し、例題のスクリプトを打ち込んでいきます。この時、後でタイプミスをしも気にしないで一気に入力します。

入力が完了したら（最後の行まで入力したら）`[ESC]` キーを押してください。

## サンプルスクリプト `wdate`

```
#!/bin/bash
export TZ
TZ=Europe/Berlin
date
TZ=US/Hawaii
date
TZ=Australia/Sydney
date
TZ=Japan
date
TZ=Asia/Shanghai
date
```

タイプの苦手な人は適当に `TZ~date` の組合せを間引いても構いません。

シェルスクリプトを動かす

スクリプトファイルができれば、次にファイルの属性を修正します。一般的なプログラミング言語の場合は、このあとコンパイルという作業があります（たとえば C 言語や、Java など）。が、しかしスクリプト（実は台本という意味があります）の場合は、ファイルの属性を修正するだけで、皆さんがいちいちキーボードからコマンド群を入力しなくても一気に実行してくれます。

具体的には `chmod` コマンドを使って、`wdate` の属性を変更します。

```
$ chmod +x wdate
```

```
$ ls -l wdate
```

```
-rwxrwxr-x 1 turbo turbo 124  3月 16 17:24 wdate
```

詳しくは、Linux ベーシックコースで解説しますが、ファイル一覧の記号の羅列に `x` が 3 回現われていれば属性は変わっています。

実行するには、ファイルをコマンドとして入力するだけです。ただし少しばかり呪文が必要で、ファイル名の前にドットとスラッシュをつけて実行します。

```
$ ./wdate
```

```
2009年  3月 18日 水曜日 11:22:56 CET
```

```
2009年  3月 18日 水曜日  0:22:56 HST
```

```
2009年  3月 18日 水曜日 21:22:56 EST
```

```
2009年  3月 18日 水曜日 19:22:56 JST
```

```
2009年  3月 18日 水曜日 18:22:56 CST
```

スクリプトの改良（1）

このスクリプトで使っている `date` コマンドは、地域(タイムゾーン)を記号で表示しているため、この出力を見ただけではどの地域かは分かりにくいですね。せっかく TZ に `Europe/Berlin`<sup>1</sup> や `Japan` と設定したのに、`CET`(Central Europe standard Time)、`JST`(Japan Standard time)と表示されています。

そこで、地域名を表示するよう `wdate` を改良しましょう。まず `wdate` をコピーし `wdate2` を作ります。

```
$ cp wdate wdate2
```

そして、`wdate2` を `vi` で開き、編集します。

```
$ vi wdate2
```

```
date
```

date 部分を下記のように修正します

```
echo "$TZ `date`"
```

少し紛らわしいですが、``` はダブルクォーテーション(キーボードは `[2/ふ]` の位置)で、``` はバッククォーテーション(キーボードは `[@/`]` の位置)です。

<sup>1</sup> 地域情報は `/usr/share/zoneinfo` というディレクトリ下に格納されています。

viらしく、小技を駆使して修正しましょう。

date の先頭にカーソルを移動し、iで入力モードになってから、echo \$TZ `までを入力し[ESC]を押します。

```
date
```

この位置で、iecho "\$TZ `[ESC]と入力する。

```
echo "$TZ `date
```

続いて、次の date の先頭にカーソルを移動します。

そこで、.(ドット)をタイプすると、一気にさっきと同じ操作が行われます。ドットは直前に実行したコマンドを再度実行するコマンドです。

```
echo "$TZ `date
TZ=US/Hawaii
echo "$TZ `date
```

 一気に入力される。

これを date があるだけ繰り返します。

```
#!/bin/bash
export TZ
TZ=Europe/Berlin
echo "$TZ `date
TZ=US/Hawaii
echo "$TZ `date
TZ=Australia/Sydney
echo "$TZ `date
TZ=Japan
echo "$TZ `date
TZ=Asia/Shanghai
echo "$TZ `date
```

date の前半部分の修正がおわったら、今度は date の末尾に閉じるためのクォーテーションを追加します。

この場合も効率的に行うには以下のように操作を行うとよいでしょう。

A` [ESC]で、末尾にバッククォーテーション、ダブルクォーテーションを追加し再びドットを使って繰り返し追加します。

## サンプルスクリプト wdate2

```
#!/bin/bash
export TZ
TZ=Europe/Berlin
echo "$TZ `date`"
TZ=US/Hawaii
echo "$TZ `date`"
TZ=Australia/Sydney
echo "$TZ `date`"
TZ=Japan
echo "$TZ `date`"
TZ=Asia/Shanghai
echo "$TZ `date`"
```

最後に :wq [Enter] または ZZ でファイルを保存します。

wdate2 は wdate をコピーして作っているのですが、すでに属性は変更されている状態になっています。実行すると、以下のようになります。

```
$ ./wdate2
US/Hawaii      2009年  3月 18日 水曜日 22:26:07 HST
Australia/Sydney 2009年  3月 19日 木曜日 19:26:07 EST
Europe/Berlin  2009年  3月 19日 木曜日 09:26:07 CET
Asia/Shanghai  2009年  3月 19日 木曜日 16:26:07 CST
Japan          2009年  3月 19日 木曜日 17:26:07 JST
```

## スクリプトの改良(2)

どうにか地域も表示されるようになりましたが、もう少し「見てくれ」を格好良く修正したいものです。かといって再び表示部分を何回も修正するのは面倒です。そこでスクリプトを修正してもう少し賢いやり方にします。

## サンプルスクリプト wdate3

```
#!/bin/bash
export TZ
for TZ in US/Hawaii Australia/Sydney Europe/Berlin Asia/Shanghai Japan
do
    city=`basename $TZ`
    printf "%10s %20s/n" "$city" "`date`"
done
```

for は do ~ done の間を繰り返します。上記の例では変数 TZ に in 移行の文字列を順に代入して、要素がなくなるまで繰り返します。つまり最初は US/Hawaii つぎに Australia/Sydney と続いて、最後に Japan となります。

basename コマンドは、続く引数をのうち、スラッシュで区切った要素の最後の値を返します。たとえば /A/B/C なら、C を返します。つまり US/, Australia/, Europe/, Asia/ を削除し、Hawaii, Sydney Berlin, Shanghai を返します。

最後に printf ですが、これは表示する際の桁数を指定しています。%10s は文字を 10 桁で表示、¥n は改行を意味します。こういった桁数や改行などの表示様式を指定することをフォーマットと呼びます。(printf は print with format の意味です)

続く引数は、basename で加工した TZ の内容と、date コマンドの実行結果をフォーマットの %10s, %20s にそれぞれ差し込みます。

%がフォーマットの様式を指定するキーワード開始を意味しています、次の数字が桁数、s が表示方法です。この場合は 10 桁の文字列と、20 桁の文字列を表示させています。

他にも %03d で前ゼロ付き 3 桁の整数、%5.2f 小数点以下 2 桁の実数などといった指定ができます。

さてスクリプトは大幅な改良ですが、これも vi の小技を使えばかなり簡単に修正できます。例によってファイルをコピーし、修正を開始しましょう。

```
$ cp wdate2 wdate3
```

```
$ vi wdate3
```

```
#!/bin/bash
export TZ
TZ=US/Hawaii
echo "$TZ      `date`"
TZ=Australia/Sydney
echo "$TZ      `date`"
TZ=Europe/Berlin
:
```

vi でファイルを開いたら、まず `:%s/TZ=//` [Enter] とします。これは全ての行について、TZ= という文字列を削除するという意味です。( : 開始行, 終了行 s/文字列 1/文字列 2/ )  
これを実行すると。以下ようになります。

```
#!/bin/bash
export TZ
US/Hawaii
echo "$TZ      `date`"
Australia/Sydney
echo "$TZ      `date`"
:
```

次に 2 回目のタイムゾーンが記載されている行(この例では Australia/Sydney)に移り、`dd`(1 行削除 = カット)、直後に 1 回目のタイムゾーンの行へ戻って `p`(ペースト)を実行します。

```
export TZ
US/Hawaii
Australia/Sydney
echo "$TZ      `date`"
Australia/Sydney
echo "$TZ      `date`"
```



これを繰り返し、`export` 文の下にタイムゾーンを並べます。

```
#!/bin/bash
export TZ
US/Hawaii
Australia/Sydney
Europe/Berlin
Asia/Shanghai
Japan
echo "$TZ      `date`"
echo "$TZ      `date`"
:
```

次に `export TZ` の行に移動し、`for TZ in` [ESC] とします。

```
#!/bin/bash
export TZ
for TZ in
US/Hawaii
Australia/Sydney
```

for 文の in の n にカーソルがある状態で、J をタイプします。J は下の行をつなぐコマンドです。

```
for TZ in US/Hawaii
```

繰り返し J をタイプすると、次々に下の行が連結されます。

```
#!/bin/bash
export TZ
for TZ in US/Hawaii Australia/Sydney Europe/Berlin Asia/Shanghai Japan
echo "$TZ      `date`"
:
```

for 文の下に do ~ done の内容を追加します。odo[ Enter ][ TAB ]city ~ done[ ESC ]

```
for TZ in US/Hawaii Australia/Sydney Europe/Berlin Asia/Shanghai Japan
do
    city=`basename $TZ`
    printf "%10s %20s /n" "$city" "`date`"
done
echo "$TZ      `date`"
:
```

最後に、不要な echo 分を削除します。最初の echo 文にカーソルを移動して、dG とタイプします。



echo 文をごっそり削除して、:wq [ Enter ] または ZZ で保存し終了します。

実行すると、以下のようになります。

```
$ ./wdate3
```

```
Hawaii 2009年  3月 19日 木曜日 20:57:44 HST
Sydney 2009年  3月 20日 金曜日 17:57:44 EST
Berlin 2009年  3月 20日 金曜日 07:57:44 CET
Shanghai 2009年 3月 20日 金曜日 14:57:44 CST
Japan 2009年   3月 20日 金曜日 15:57:44 JST
```

フォーマットの %10s を %-10s とすると、文字列は左寄せになります。

```
$ ./wdate4
```

```
Hawaii   2009年  3月 20日 金曜日 05:44:21 HST
Sydney   2009年  3月 21日 土曜日 02:44:21 EST
Berlin   2009年  3月 20日 金曜日 16:44:21 CET
Shanghai 2009年  3月 20日 金曜日 23:44:21 CST
Japan    2009年  3月 21日 土曜日 00:44:21 JST
```

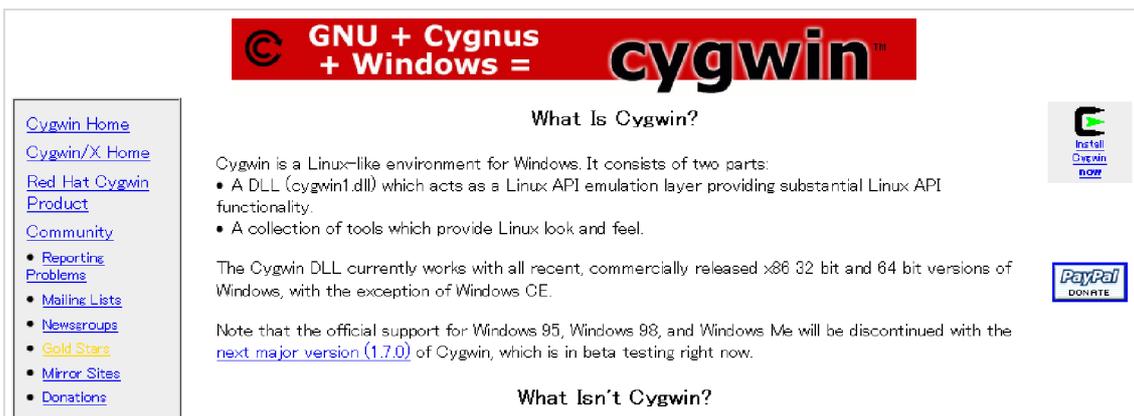
## 付録 1 : 主な vi コマンド一覧

コマンド	意味
<b>挿入コマンド群</b>	
A	行末に追加 (Append)
I	行頭に挿入 (Insert)
a	カーソル直後に追加
i	カーソル直前に挿入
O	カーソルの上に 1 行加える (Open)
o	カーソルの下に 1 行加える
<b>カーソル移動コマンド群 (1)</b>	
k, [ ], -	一行上へ移動
j, [ ]	一行下へ移動
h, [ ]	一文字分左へ
l, [ ], . [ ]	一文字分右へ
0(ゼロ)	行の先頭へ
\$	行末へ移動
, +	下の行の、非空白文字へ移動
n  (縦棒)	今いる行の (n)文字目に移動
nG	n行目に移動。nを省略すると最終行へ移動
:n	n行目に移動。nに \$を指定すると、最終行へ移動
<b>カーソル移動コマンド群 (2)</b>	
w	現在位置の右にある単語の先頭にカーソルを移す。
B	現在位置の左にある単語の先頭にカーソルを移す。
E	現在位置の右にある英字または特殊文字が連続する最後の文字にカーソルを移す。
W	現在位置の右側にある空白で区切られた単語の先頭にカーソルを移す。
B	現在位置の左側にある空白で区切られた単語の先頭にカーソルを移す。
E	空白で区切られた単語の終りにカーソルを移す。
%	対応する括弧への移動。( と ) { と } など
<b>テキストの削除コマンド群</b>	
x	カーソル下の 1 文字を削除
X	カーソルの左側 1 文字を削除
dd	カーソルのある行を削除
D	カーソルから行末までを削除
d移動	カーソル位置から移動コマンドで移動したところまでを削除
<b>テキストの修正コマンド群</b>	
cc	カーソル位置の行を、エスケープまでの入力文列と置き換える。
r文字	カーソル位置の 1 文字を、文字 に置き換える。
c移動	カーソル位置から移動コマンドで移動したところまでを修正。
~	カーソル位置の大文字 / 小文字の置き換え
:%dstr1/str2/g	ファイル全体の文字列 str1を str2に置き換える。
<b>その他</b>	
u	直前のコマンドを取消し、もとの状態にもどす (Undo)。
yy	1 行コピー (Yank)
p	yy または dd した行をペースト
.	直前の入力操作を繰り返す
J	下の行を連結する (Join)
<b>ファイル操作</b>	
:w file	現在の内容をファイル fileとして保存
:r file	現在の内容に fileの内容を追加取り込み。
<b>終了</b>	
ZZ	内容を保存して終了。( :wq[Enter]も可 )
:q!	保存せずに終了
:w	今の内容をファイルへ上書き
:e!	ファイルの取り込みなおし (今の編集内容を破棄し最初から)

## 付録 2 : Cygwin

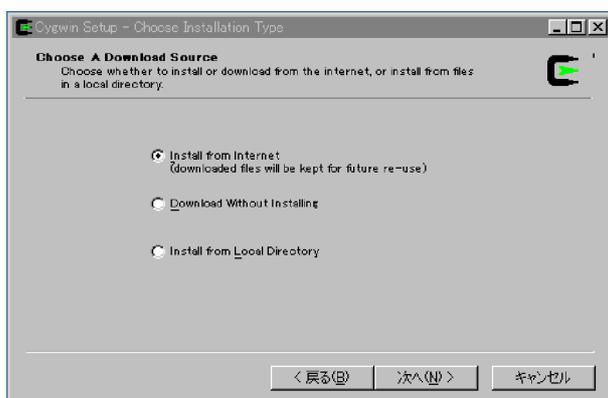
Cygwin(シグウィン)は Windows 上で UNIX の環境を実現するエミュレータの一つです、ホームページからセットアッププログラムをダウンロードし実行すると、ネットワーク経由で必要なパッケージを設定できます。

<http://www.cygwin.com/> にアクセスし、キットをダウンロードします。右上にある E に似たアイコンをクリックしファイルをダウンロードしてください。



起動すると確認画面が表示されます。

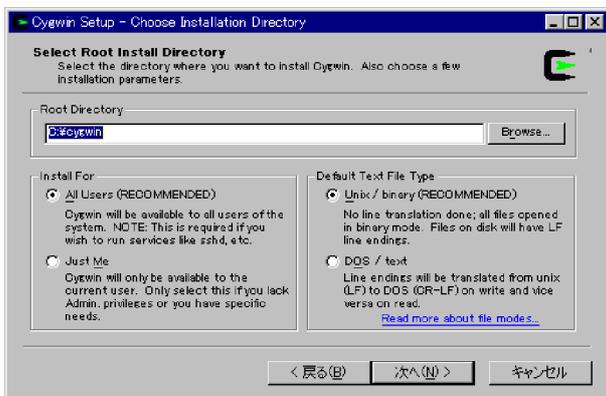
現在の Cygwin は 1.5 が最新です。この画面で表示されているバージョンは、セットアッププログラムのバージョンはですので間違わないよう気を付けてください。



Cygwin 本体をどこからダウンロードするかを指定します。

Install from Internet

を選択し、インターネットから本体をダウンロードするよう指定します。



インストール先を指定します。どこでも良いのですが、パス名（フォルダ名）には日本語や空白を含まないようにします。残りの設定は、省略値でかまいません。



Cygwin のパッケージ(追加機能)を格納するパスを指定します。ここもパス名に日本語や空白を含まないように指定してください。

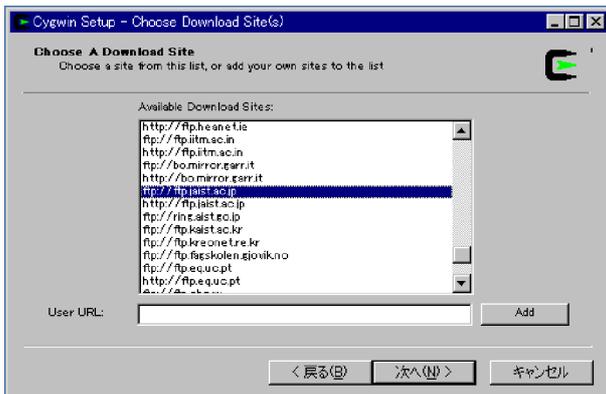


インターネットに接続する方法を指定します。自宅の場合は

Direct Connection

を選択します。

企業などのネットワークの場合はプロキシと呼ばれる、外部ネットワークとの接続を仲介するサーバーが用意されていることがあります。

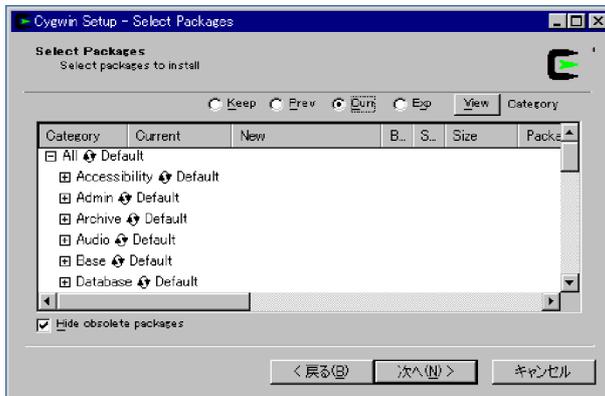


Cygwin のファイルサーバを指定します。日本からのアクセスの場合は国内のサーバを選択します。

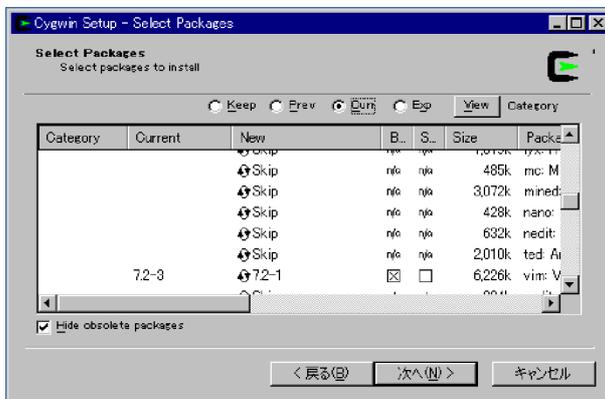
<http://ftp.jaist.ac.jp>

<ftp://ring.aist.go.jp>

など.jp で終わる URL を選択します。



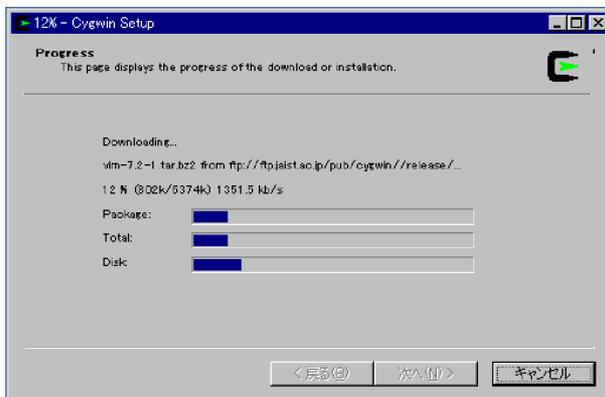
インストールするパッケージを選択します。パッケージは大まかな機能で分類されており、かなりの種類があります。



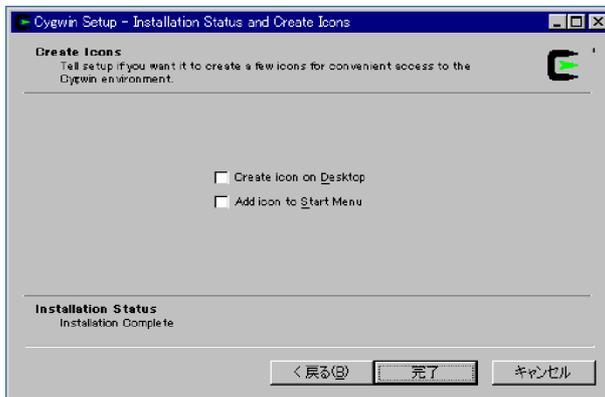
カテゴリの Editor をクリックすると、いろんなエディタが一覧として表示されます。

その中から vim を探し、行をクリックします。クリックするたびにインストールしない(Skip)、インストールする(バージョン表示)が切り替わります。

インストールを選択してください。



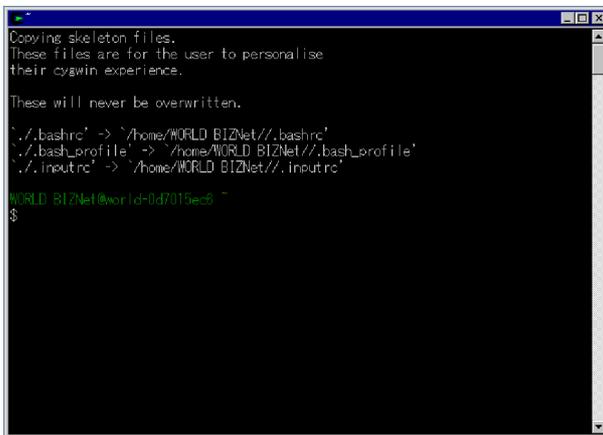
次へ進むと指定したファイルサーバーからダウンロードが開始され Windows に必要なファイル群がインストールされます。



最後にデスクトップとメニューに Cygwin のアイコンを追加するかどうかの確認画面が表示されます。

チェックをして、完了をクリックします。

デスクトップにある Cygwin アイコンをダブルクリックするか、スタートメニューから Cygwin Bash Shell を選択して Cygwin を起動します。



```
Copying skeleton files.
These files are for the user to personalise
their cygwin experience.

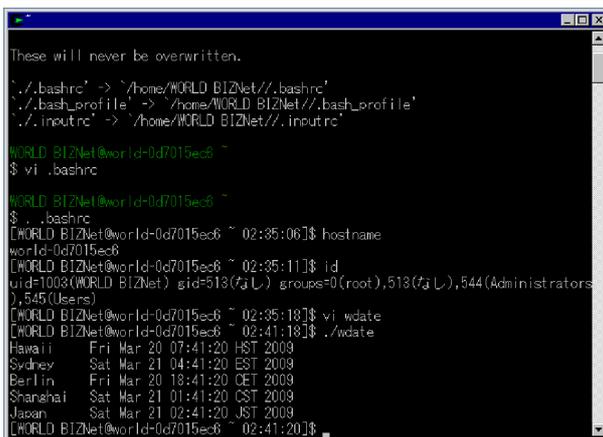
These will never be overwritten.

./.bashrc' -> '/home/WORLD_BIZNet//.bashrc'
./.bash_profile' -> '/home/WORLD_BIZNet//.bash_profile'
./.inputrc' -> '/home/WORLD_BIZNet//.inputrc'

WORLD_BIZNet@world-0d7015ec6 ~
$
```

Cygwin を起動すると Windows のコマンドプロンプトとよく似た画面が表示されます。

この黒い画面の中は、ほぼ Linux のコマンドが利用できます。



```
These will never be overwritten.

./.bashrc' -> '/home/WORLD_BIZNet//.bashrc'
./.bash_profile' -> '/home/WORLD_BIZNet//.bash_profile'
./.inputrc' -> '/home/WORLD_BIZNet//.inputrc'

WORLD_BIZNet@world-0d7015ec6 ~
$ vi .bashrc

WORLD_BIZNet@world-0d7015ec6 ~
$ cat .bashrc
[WORLD_BIZNet@world-0d7015ec6 ~ 02:35:06]$ hostname
world-0d7015ec6
[WORLD_BIZNet@world-0d7015ec6 ~ 02:35:11]$ id
uid=1003(WORLD_BIZNet) gid=513(なし) groups=0(root),513(なし),544(Administrators),545(Users)
[WORLD_BIZNet@world-0d7015ec6 ~ 02:35:18]$ vi wdate
[WORLD_BIZNet@world-0d7015ec6 ~ 02:41:18]$ ./wdate
Hawaii    Fri Mar 20 07:41:20 HST 2009
Sydney    Sat Mar 21 04:41:20 EST 2009
Berlin    Fri Mar 20 18:41:20 CET 2009
Shanghai  Sat Mar 21 01:41:20 CST 2009
Japan     Sat Mar 21 02:41:20 JST 2009
[WORLD_BIZNet@world-0d7015ec6 ~ 02:41:20]$
```

このコースで試した内容は動作確認済です。

Cygwin のプロンプトには改行が入っていて使いにくいのでシェルの初期設定ファイル `.bashrc` を修正し、末尾に以下を追加するとよいでしょう。(これは Linux Basic コースの例題と同じ内容です)

```
PS1=[ \u@\h \W /t ] / $
```

- ¥u ユーザー名
- ¥h ホスト名
- ¥W 現在のディレクトリ(basename 表記)
- ¥t 現在の時刻
- ¥¥\$ \$ は特殊な意味を持ちますので、¥¥\$で\$を意味します。

## 付録 3 : Vim

Linux 環境を準備しなくても、vi の練習をすることができます。vim のホームページにアクセスし、Windows 版をダウンロードしてください。

The screenshot shows the Vim website homepage. At the top, there is a navigation bar with links for SPONSOR, VOTE, BUY, HELP, and LEARN. The main content area is divided into several sections. The 'News' section contains a link to 'Vim 2009 desktop calendar'. The 'Vim 7.2' section contains a link to 'download page', which is highlighted with a red box. The 'Recent Script Updates' section contains several links to scripts and updates. The right sidebar contains links for 'Python Editor', 'Free Web IDE Download', 'HTML Editor for ASP.NET', and 'Code Refactoring'.

ダウンロードページに「PC: MS-DOS and MS-Windows」の記述がありますので、それをクリックし。必要なファイルをダウンロードします。Self-installing executable はセットアップ形式のファイル、Runtime files は解凍するだけで利用できます。

The screenshot shows the 'PC: MS-DOS and MS-Windows' section of the Vim website. The text describes the availability of different versions of Vim for MS operating systems. It includes links to 'the pc directory', 'gvim##.exe', and 'gvim72.exe'. It also mentions that the installer allows you to skip parts you don't want and provides a note about a false alarm from Sophos Antivirus. The section concludes with a link to 'Cream' for the latest version and a link to 'Runtime files'.