

# Linux の個人 PC へのインストール

プログラミング演習 1 #5  
平成 26 年 5 月 16 日

## 1 Linux と Linux distribution

本日の授業では、CentOS とはまた一味異なる Linux ディストリビューションである Ubuntu Linux の個人 PC へのインストールを行います。それはいいんですが、そもそも Linux ディストリビューションって何じゃらホイ、と思った方も居るかと思います。その他にも、オペレーティングシステムとか Linux とかいう言葉をこれまで漠然と使用してきました。この機会にこの辺の言葉を整理しておこうと思います。

オペレーティングシステム (OS) というのは、コンピュータを操作するための基本的な機能を提供するソフトウェアです。もちろん、どこからどこまでが「基本的な機能」か、というのは意見がわかれるところで、同様に、どこからどこまでが OS なのかも意見がわかれるところです。が、ここでは、アプリケーションソフトウェア以外のソフトウェアの部分全てをオペレーティングシステムと呼ぶことにします。このとき、ハードウェアを扱う中心的な機能を実現する部分をカーネル、それ以外の部分をユーザーランドと呼びます。(「OS とはカーネルの部分のことを指す言葉である」と考えている人もいます。)

Linux というのは、元来はカーネルの 1 種で、自由なライセンスのもとで無料で公開されています。自由なライセンスのもとで公開されているカーネルとしては、Linux の他にも Hurd や FreeBSD などがあります。

ユーザーランドを構成するソフトウェアに関して、多くのものが GNU プロジェクトなどの団体によって作成され、同様のライセンスで無料で公開されています。このため、カーネルやユーザーランドを好きなように組み合わせて、動作確認を行うことにより、様々な OS を作成することができます。このようにして作られた OS のうち、Linux をカーネルに使用しているものを Linux ディストリビューションと呼びます。主要な Linux ディストリビューションとしては、Debian GNU/Linux, Ubuntu Linux, RedHat Linux, Fedora Core, Slackware, Gentoo Linux などがあります。Debian GNU/Hurd, Debian GNU/kFreeBSD など、同じユーザーランドでカーネルを変えた OS もあります。

## 2 Ubuntu Linux について

Ubuntu Linux は、Debian から派生した Linux ディストリビューションの 1 つです。2004 年 10 月の最初のリリース当時以降、ほとんど常に DistroWatch.com の人気ランキング 1 位の座を占めるほどの圧倒的人気を誇っています。

元となっている Debian 自体、元来非常に優れたディストリビューションだったのですが、プロジェクトのポリシーが完璧主義的で、いつリリースされるかわからない点が不満の種になっていました。加えて、インストールのしにくさも敬遠される理由の一つになっていました。Ubuntu Linux は、「半年ごとに新しいバージョンがリリースされる」ことを約束することで前者の問題を解決<sup>1</sup>、また Debian ベースのライブ CD(3.1 節で後述)である Knoppix からインストーラやハードウェア認識システムを流用することで後者の問題を解決した、「話の分かる Debian」であるといえます。

本学科の演習室では、CentOS という Linux ディストリビューションを使うことになっています。が、この CentOS 自体非常に更新が遅い上に、演習室の環境の更新自体頻繁に行えないため、比較的古いソフトウェアパッケージしか入っていないで非常に残念です(しかも、皆さんが 1 年の頃に慣れ親しんだ環境は、皆さんが

<sup>1</sup>といってもこの約束は 2006 年 4 月リリース予定だった Dapper Drake においてあっさり破られたわけですが。

社会に出る頃にはさらに古くなっています。) また、学生の皆さんに管理者権限が与えられていないため、最新のソフトをインストールしたりシステム管理を学んだりすることができません。そこで、今日の演習ではこの Ubuntu Linux を皆さんの個人 PC にインストールする方法を学びます。

とりあえず Ubuntu Linux の長期サポート版 (14.04 LTS, Trusty Tahr) の 32bit 版の CD と 64bit 版の CD を用意しておきました<sup>2</sup>。どちらを選択するのがよいかは一概には言えませんが、

- 32bit CPU では 64bit 版は動かないので、その場合は 32bit 版を使うことになる。使っている PC の CPU が 32bit か 64bit かを判断するの単純ではないですが、
  - インストールされて稼働している Windows が 64bit 版なら 64bit であることが確定する。32bit 版の場合、プロセッサ名をググって調べる。てか、Core なら 64 ビット、Atom なら 32 ビットで間違いなかったと思う。  
プロセッサ名や、Windows が何 bit 版かは、スタート コントロールパネル システムとセキュリティ システム で分かる。
  - 64bit 版のライブ CD から起動して動作すれば 64bit. 動作しなければ 32bit
- 64bit CPU では、基本的には 64bit 版のほうが早く動作するはず (もともと、Flash に限って言えば 32bit 版の方が速いという噂がある。)

という点を考慮に入れた上で判断すればよいでしょう。

## 2.1 Ubuntu のフレーバー

Linux のインストール CD (やライブ CD) は容量に限りがあります。また、DVD を使うにしても、データ量が無駄に多いとダウンロードに余計な時間がかかります。そこで、Ubuntu Linux では、1つのインストールメディアに対して1つのデスクトップ環境しか用意していません。そのかわり、色々なデスクトップ環境のインストールメディアを用意しています。これらの個々のデスクトップ環境に対応するものをフレーバーと呼びます。また、デスクトップ環境以外にも色々なソフトウェアをバンドルしたメディアもあり、これらもフレーバーと呼ばれます。表 1 に主なフレーバーを載せます。

表 1: 主な Ubuntu のフレーバー

フレーバー	デスクトップ環境	パッケージ名
Ubuntu	Unity (タッチパネル向け)	ubuntu-desktop
Kubuntu	KDE (高機能)	kubuntu-desktop
Ubuntu GNOME	GNOME	ubuntu-gnome-desktop
Xubuntu	XFCE (軽量)	xubuntu-desktop
Lubuntu	LXDE (軽量)	lubuntu-desktop

インストール後のデスクトップ環境の追加は、インターネット接続がある環境であれば、非常に簡単に行えます。具体的には、

```
sudo apt-get install パッケージ名
```

<sup>2</sup>正確には、Lubuntu と呼ばれるフレーバー (後述) となります。

というコマンドを実行して管理者パスワードを入力すれば、あとは自動でやってくれます。という訳で、普通にインストールして使う分には、どのフレーバーのメディアを使っても変わらなかったりします。

ただし、この1年ほどで Ubuntu のインストールディスクの容量が増大し、Lubuntu 以外は1枚のCDに収まらなくなってしまいました。<sup>3</sup>そこで皆さんには、Lubuntu のCDを配布することにします。

## 3 1台のPCで複数OSを使用する方法

本日の授業では、Windows がインストールされているPC上に Windows と共存する形での Ubuntu Linux のインストールを行います。まず、1台のPCで複数OSを使用する方法についていくつか紹介します。

この授業で扱うのは、簡単安全お手軽な仮想化によるものです。パーティション切り分けによる方法は、操作を誤るとハードディスク上のデータを破壊してしまう可能性があるため、自己責任でお願いします。なお、破壊してしまったデータの復活に商用サービスを用いると、通常万あるいは十数万円単位の値段がかかります。ま、余程価値のあるデータでなければ復活は諦めて再インストールすればいいんですが、価値があるデータがある場合は予めUSBメモリなどにバックアップを取っておきましょう。

なお、既に Ubuntu をインストールしている強者がいるなら、その人は今回はすることないかも。

### 3.1 ライブCD/ライブDVD/ライブUSBメモリ

多くの場合において、ハードディスクや内蔵フラッシュメモリ(SSD)にインストールされたOSを利用するのが一般的です。しかし基本的には、OSといえども単なるプログラムに過ぎず、従ってOSを利用する上で、OSを格納する場所はあまり問題ではありません。実際、演習室のシステムにおいては、ネットワークの向こうにあるサーバに保存されたOSを手元のコンピュータのメモリにコピーしてきて使用しています。

ハードディスクや内蔵フラッシュメモリの代わりに、CD-ROMなどにOSをインストールしたものをライブCDと呼びます。また同様に、DVD-ROM等にインストールしたものをライブDVD、USBメモリにインストールしたものをライブUSBメモリと呼びます。(以下、これらをまとめて「ライブCD等」と呼びます。)

#### 3.1.1 起動方法

ライブCD等からOSを起動するには、ライブCDなりDVDなりをドライブに挿入した状態で(またはUSBメモリをUSBポートにさした状態で)コンピュータを再起動すればよいです。ただし、ハードディスクとCDとUSBメモリのうち複数の場所にOSが入っている場合にハードディスクを優先する設定になっている場合は、この設定を変える必要があります。この優先順序はブートシーケンスと呼ばれ、ブート画面(コンピュータを起動した直後の画面)で指示されたキーを押すことで設定画面に入ることができます。

#### 3.1.2 特徴

ライブCD等を用いると、ハードディスクにインストールされているOSとは全く関係ないOSを、ハードディスクを全く変更することなく走らせることが可能です。このため、OSのお試し版を配布したり、動かなくなってしまったOSを修復したりそこからデータを回収したり、犯罪の証拠となるハードディスクから非破壊でデータを抽出するのに使われます。近年では、LinuxのインストールメディアがライブCD等を兼ねていることが多く、

<sup>3</sup>個人的には、どうせDVDにするなら全部入りのものを用意してもいいんじゃないかとは思っていますが、取り敢えずそういうのはなさそうです。

1. 最初にライブ CD 等でいろいろ試した後,
2. 気に入ったらインストールアイコンをクリックして(パーティション切り分けによる) インストール(後述)

という手順で便利にインストールすることができます。

一方で、ライブ CD 等を用いる場合、ホームディレクトリは通常メモリ上に置かれるため、設定や作成した文書などは、特になにも対策しないと再起動時に消えてしまいます。このため、一時的な利用以外では、ライブ CD 等を用いるよりも、ちゃんとハードディスクにインストールした方が便利なが多いです。

## 3.2 パーティション切り分けによる方法

古来(?) からある最も自然かつオーソドックスな方法は、パーティション切り分けによるものです。パーティション分けというのは、1 台のハードディスクを複数の部分(パーティション)に分割して使用することです<sup>4</sup>。たとえば、Windows と Linux を共存させる場合、最も単純な方法は、

1. 1 台のハードディスクの前半(/dev/sda1)を Windows 用、後半(/dev/sda2)を Linux 用に見立てて、
2. 前半を Windows の標準的ファイルシステム(ファイルやディレクトリの整理方法)である NTFS、後半を Linux の標準的ファイルシステムである Ext4 にし、
3. 前半に Windows を置いて、後半に Linux を置く

という構成になります。

### 3.2.1 起動方法

電源を入れた後、あるいは再起動後、起動する OS を選択する画面(ブートローダ画面)になります。要は演習室のシステムと同じ選択方法になります。(ただし大抵は、演習室の選択画面ほどそっけなくはないですが。)

### 3.2.2 特徴

新しくインストールした OS と古い OS は対等な立場でハードディスク上に共存することになります。このため、他の方法と異なり、新しくインストールした方の OS に機能制限がなく、また動作速度を犠牲にすることなく使用することができます。

作業中に OS を切り替えるには、一旦休止状態(hibernation, suspend to disk)に入ってから、起動し直すことになります。このため、仮想化による方法に比べると切り替えに余計な時間がかかります。

Windows と Linux を共存させる場合において、Windows を再インストールしたりアップグレードしたりすると、Windows が勝手にブートローダを書き換えて Linux が起動しなくなることがあります。その場合はたとえば <http://marumarusan003.blog8.fc2.com/blog-entry-156.html> あたりを参考に修復すると良いです。

---

<sup>4</sup>複数台のハードディスクをつないでいる場合はパーティション分けなしに同様の方法でインストールできます。

### 3.2.3 インストール手順

Windows 8 が出る前のパソコンであれば、殆どの作業はインストーラーが自動的にやってくれます。Windows 8 が出てきて、BIOS が UEFI になって、とたんにめんどくさくなりました。ここでは、Windows 8 が出る以前の PC でならうまくいく方法を書いておきます。最近の PC でもだいたいの手順は同じなんですが、機種によって個別のおまじないが必要だったりしてハマるので、まあやめといたほうがよいです。

1. Ubuntu の CD をドライブに挿入し、PC を再起動します。
2. Ubuntu 14.04 LTS のインストールのアイコンをダブルクリック
- 3.あとは適当に指示に従っていきます。

途中、「インストールの種類 (Installation type)」のダイアログになります。ここでパーティションの切り方を指定します。よくわからない人は、「Ubuntu を Windows とは別にインストール / Install Ubuntu alongside Windows 7」が選択するのが無難です。

### 3.2.4 アンインストール方法

標準的なアンインストール方法というのはなく、

- ブートローダの再インストール
- パーティションの削除 (残す方のパーティションへの統合)

を行えばアンインストールしたことになります。前者は、Windows を残すのであれば Windows の再インストールによって可能です。後者は、ライブ CD 等で起動して gparted や qtparted などのパーティションエディタで実行可能です (というか、Windows も 7 辺りから標準でパーティションエディタを持っているようなので、そちらでやったほうがお手軽かもしれません。これについてはあまり詳しくないのでゴニョゴニョ)

が、パーティションエディタを用いて操作を間違えたらうっかりディスク全体を削除してしまうこともありえるので、初心者にはおすすりめできません。どうしてもパーティションを削除したい場合は、その前に重要なデータのバックアップを取っておきましょう。

## 3.3 Wubi

Wubi (Windows-based Ubuntu Installer) は、Windows のインストールされている PC に Ubuntu Linux をインストールする方法の一つです。パーティションを切り直す代わりに、Windows (NTFS) 上の 1 つの大きなファイルを作って、その中に Ubuntu 用のファイルシステムを構築するものです。

Windows 7 の時代までは、安全お手軽かつ安定的にインストールする方法として Linux 初心者にお薦めの方法だったのですが、Windows 8 になって色々ややこしくなり、本来のメリットが無くなってしまいました。Windows 7 を使っているけどパーティションを切り分けるのはちょっと怖い、という方にはお勧めです。

なお、同じような方法で Linux 上に Ubuntu Linux をインストールする方法として Lubi というものもあります。

### 3.3.1 起動方法

パーティション切り分けによる方法と同様に、ブートローダ画面で選択します。ただし、Ubuntu のインストール・アンインストールを Windows 側が管理している関係上、まず Windows のブートローダが立ち上がることとなります。とはいえ、Linux カーネルのバージョンを起動時に選択する目的で Linux のブートローダも必要であるため、

1. Windows のブートローダが起動し、
2. これが Linux のブートローダを起動し、
3. これが Ubuntu Linux を起動する、

という多少まわりくどい順番になります。

### 3.3.2 特徴

インストール時、アンインストール時にパーティション操作が不要であるため、初心者にとって安心です。それだけで、仮想化に比べると動作速度のロスが少なく安定的です。

欠点としては、

- 休止状態に入れない。このため、Windows に切り替えるには一旦 Linux を終了しなければならず、後で Linux に戻って来た時に、中断したところから再開できない。
- 電源障害などでファイルシステムが壊れた時、修復が困難である。
- パーティション切り分けによる方法と比べて、ハードディスクへのアクセスが少し遅くなる。

という3点が挙げられます。

パーティション切り分けによる方法に対する利点としては、「インストール、アンインストール時に楽かつ気楽」という一時的なものしかない一方で、使い続ける上での明らかな欠点がいくつかある事になります。このため、ある程度パーティション切り分けに慣れている人であれば、Wubi を使っても特になにも嬉しいことはなさそうです。

### 3.3.3 インストール手順 (CD を用いる方法)

Wubi によるインストールでは、`wubi.exe` というプログラムを実行します。`wubi.exe` には、Linux の CD イメージファイルをネットワークからダウンロードする機能や、手元にある CD イメージファイルを用いてインストールする機能、および、手元にある CD からインストールする機能があります。ここでは、CD からインストールする方法を扱います。

1. Windows を起動してください。
2. ネットワーク接続をすべて切断して下さい。これは学内でインストールするときは必ず行なって下さい。最近のノートパソコンでは、大抵無線ネットワークを切断するためのスイッチがついています。

`wubi.exe` は、インストールしようとする Ubuntu のフレーバーと CD 上の Ubuntu のフレーバーが異なる場合、カレントディレクトリに所望のイメージファイルがなければ、Bittorrent を用いて自動的にネットワークからダウンロードしようとします。しかし、学内ネットワークにおいて、Bittorrent の使用は禁止されています。

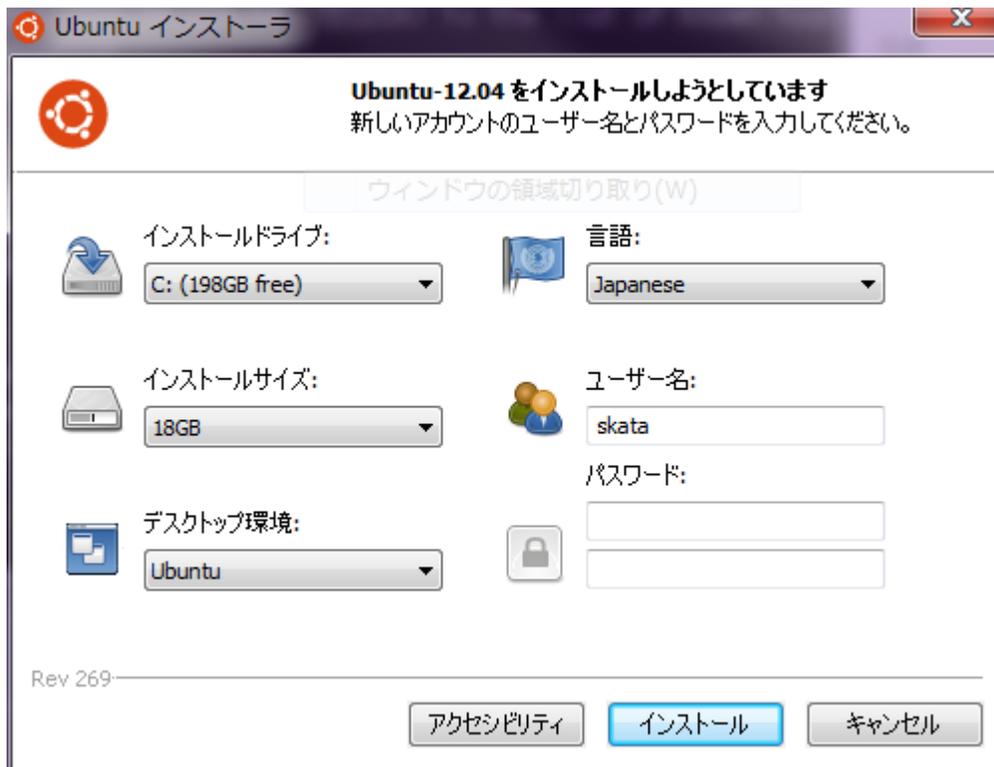


図 1: Wubi

3. Ubuntu の CD をディスクドライブに挿入し、エクスプローラで開いて、CD に入っている `wubi.exe`<sup>5</sup> をデスクトップかどこかにコピーします。
4. コピーした方の `wubi.exe` を起動します。ちなみに、CD 上の `wubi.exe` をコピーせずに直接実行すると、別の表示になってしまうようです。
5. (既に Wubi でインストールした Ubuntu がある場合、アンインストールするか聞いてきます。深く考えずにアンインストールを選んでしまうと、これまでに Ubuntu のファイルシステム上に作ったファイルは一瞬で全て消えてしまい、復活は非常に困難になるので注意が必要です。)
6. 図 1 のようなダイアログボックスが表示されます。インストールする Ubuntu のデスクトップ環境が CD と同じであることを確認し、その他の部分を必要なら適当に変更し<sup>6</sup>、新しく作成するアカウントの情報を入力して、インストールを開始します。
7. インストーラはファイルのコピーを開始します。ハードウェアの性能によりますが、CD からのコピーは早ければ数分で終わることと思います。(手元のマシンでは 5 分)
8. コピーが終わると、CD が eject されて、すぐに再起動するか聞いてくるので、セーブすべきものをセーブして再起動しましょう。

なお、この段階で、ネットワークのスイッチを再びオンにして OK です。

<sup>5</sup>設定によっては、単に `wubi` と表示されているかもしれません。

<sup>6</sup>私見を言えば言語は English が無難だと思いますが、その辺は好みで決めて下さい。

9. 再起動すると、ブートローダの画面になって Windows か Ubuntu かを聞いてくるので、矢印キーで Ubuntu を選んで Enter を押します（キー入力せずにのんきに待っているとしばらくして Windows が起動されてしまうので注意。）さらに Linux カーネルのバージョンを聞いてきますが、これは最初から選ばれているもののままで Enter を押して下さい。
10. さらにインストールが続きますが、しばらく（手元のマシンでは 10 分）するとログイン画面になります。

### 3.3.4 アンインストール手順

通常の Windows アプリケーションと同等の方法です。つまり、コントロールパネルから「プログラムのアンインストール」を選択し、「Ubuntu」を検索して選択した後「アンインストールと変更」をクリックするとダイアログボックスが開くので、そこで「アンインストール」をクリックして下さい。

ただし、通常の Windows アプリケーションと違い、作成したファイルや設定なども削除されてしまうので、必要であればバックアップを取っておきましょう。

## 3.4 仮想化

VMWare Player や VirtualBox などの仮想化ソフトウェアは、仮想上の機械の環境を用意していて、OS などのプログラムをまるごとその上で実行することができます。たとえば、Windows 版の VirtualBox を起動して、その中の仮想機械上に Linux 環境を作ってしまうと、両方の OS を同時に実行することが可能です。

### 3.4.1 起動方法

仮想化ソフトウェアを起動し、ソフトウェアごとの指示に従います。

### 3.4.2 特徴

同じパソコン上で複数の OS を同時並行的に実行することが可能です。また、それらの OS の間でクリップボードを共有したり、ディレクトリを共有したり、ネットワークを通じて通信したりできます。

一方で、複数 OS を同時に実行するということは、逆に言うと 1 台あたりの処理速度の低下を意味します。さらに、仮想環境上ではどうしてもオーバーヘッドが出てしまうため、その理由からも本来の効率を発揮できないこととなります。また、物理メモリもそれぞれの OS が別々に使用するため、たとえば 4GB のメモリがないと使っていてイライラする、という人が満足に両方の OS を使うには、合計で 8GB 必要となります。

### 3.4.3 インストール方法

たとえば、Windows 版の VirtualBox を起動して、その中で Linux のインストーラを走らせることで、仮想機械上に Linux 環境を作ることができます。複数の OS をインストールする場合、OS ごとに仮想機械を用意します。

Windows を削除して代わりに Linux をインストールしておいて、Linux 版の VirtualBox を起動してその中に Windows 環境を作ることもできます。ただし、そういうのは自己責任でお願いします。

### 3.4.4 アンインストール方法

仮想化ソフトに仮想機械を削除するよう指示します。

## 4 VirtualBox による仮想化

今回は VirtualBox による仮想化を行います。他に有名な仮想化ソフトとしては VMWare がありますが、経験的に言って VMWare はあまり安定していないので、やめときます。

以下に手順を示します。基本的には口頭で説明するのを見ておけば良いです。

1. 32 ビット版にするか 64 ビット版にするか決める。なお、Wubi やパーティションを切る方法や Live CD 等の場合、64 ビット版の Ubuntu が動くかどうかは CPU によって決まりましたが、VirtualBox は Windows 上で動くため、64 ビット版の Windows を使っているかどうかが大きく影響します。64 ビットプロセッサでない当然 64 ビット版の Ubuntu は動きませんが、64 ビットプロセッサで 32 ビット Windows を使っている場合でも、64 ビット版 Ubuntu を走らせるにはプロセッサが hardware virtualization をサポートしている必要があります。
2. 64 ビット版を使う場合、PC を起動する前に BIOS 設定で hardware virtualization の項目 (“Intel(R) Virtualization Technology” かなんか) を Enable にしておくといよい。“BIOS ってなに?” という人とか “もう起動しちゃった” って人は取り敢えず先に進んで OK。
3. VirtualBox の Windows 版をダウンロードしてきてインストールする。基本的には、Next やら Yes やら Install やら Finish やらをクリックしていけば OK。
4. VirtualBox を起動する。
5. Ubuntu の CD を挿入しておく
6. 「新規」ってのをクリックし、新しい仮想機械を作る。テキトーな名前を入力して、Linux, Ubuntu を選択。「新規」に関してはそんなもん。ここで、64 ビットを選択したいのに 32 ビットしか選択できない場合は、次を順に試す：
  - (a) Hyper-V が有効化されている場合は無効化する。<http://blog.yoshikawa.64kb.net/2013/12/29/1229> を参考にするとよい。
  - (b) BIOS の hardware virtualization の項目が無効なら有効化する。BIOS 設定のやり方は機種によって違うのでぐぐってみてください。
7. 「設定」をクリックして、この仮想機械の設定を行う。取り敢えず必要なのは、今回はネットワーク負荷の軽減のため実際の CD からインストールするため、実際の CD を利用する設定にする。このためには、出てきたダイアログボックスについて、
  - 「ストレージ」で「コントローラ:IDE」の下の「空」の部分を選択
  - その下の「属性」以下の CD のマークをクリックして「ホストドライブ'D:」みたいなものを選択。
8. 「起動」をクリックして仮想機械を起動し、その中でインストールの作業を行う。  
Ubuntu の CD が起動する。インストール CD とライブ CD を兼ねているので、どっちを実行するか聞いてくる。ライブ CD を選んでもインストールは可能なので、どちらを選んでもよい。  
選択肢については、

- インストール中にアップデートのインストールするか聞いてくるが、みんなで一斉にやるとネットワークに負荷がかかるので、基本的にはアップデートのインストールはここでは行わない。家がお金持ちで家からインターネットに接続できる場合は家でやる。そうでない場合もこの授業中にはやらない。
- 仮想マシンのハードディスク上に何もないので、「OS が認識されませんでした」的なメッセージがでて、ディスクの中身を削除するか聞いてくる。削除してよい。
- メモリはデフォルトだと少なめの割り当てになっているが、ケチケチせず物理メモリの半分くらい割り当てたほうがよい。
- 仮想マシンの中でのみ実行するので、log in automatically を選択しておいてもいいかもな。と思ったけど、それだとあとから LXDE 以外に変更するときに面倒かも。

インストールが終わると、ディスクを抜いてエンターキーを押すことで仮想機械を再起動するよう促される。理論上はエンターキーを押せば再起動するはずだが、一旦電源 OFF して起動しなおしたほうがいいかも。

起動すると画面が変になるが、Control-Alt-F7 を押すとログイン画面になるはず。

9. あと、guest addition っつてのをゲスト（中身）OS 側にインストールしておく、クリップボードやフォルダの共有やウィンドウのシームレス化ができて楽しいのだが、疲れたので今回はパス。<sup>7</sup>

## 5 インストール直後の設定

ここでは、インストール直後に最低限やっておくべき設定を説明します。

### 5.1 言語サポートパッケージのインストール

日本語環境を整えるには、言語サポートパッケージをインストールすると良いです。実際には、これはそんなに急いでやる必要はないんですが、せっくなので実行しておきましょう。ただし、その前に、日本語の言語パック (language-pack-ja) をインストールしておく必要があります。ついでに、かな漢字変換のためのインタフェースである ibus を利用するように設定します。

1. 端末エミュレータを立ち上げます。このためには、メニューの Accessories の下の LXTerminal を実行するか、System Tools の下の UXTerm か XTerm を実行します。
2. 端末エミュレータ上で、

```
sudo apt-get install language-pack-ja
```

を実行します。パスワードを聞かれたら入力して下さい。なお、これは、「language-pack-ja というソフトウェアパッケージをインストールしてね」という意味です。

もしエラーになったら、

<sup>7</sup>sudo apt-get update; sudo apt-get upgrade; sudo apt-get install dkms 後、デバイス 仮想 CD/DVD の選択から、c:\Program Files\Oracle\VirtualBox\VBGuestAdditions.iso を選択することによってこの CD イメージを仮想的に挿入。Dash に CD のアイコンが現れるので、それをクリックすると CD がマウントされる。ターミナルから cd /media/VBOX\*/;sudo sh ./BoxLinuxAdditions.run を実行。仮想機械を一旦終了して再起動。

```
sudo apt-get update
```

を実行してから再度試すとよいです。

3. 「このアクションをすぐに実行する」というボタンをクリックすると、ネットワークアップデートを促してきますが、みんなで一斉にやるとネットワークが遅くなるのでとりあえず cancel しておきます。
4. 「言語サポートが完全にはインストールされていません (The language support is not installed completely.)」という表示になったらとりあえずインストールをクリックしておきましょう。パスワードを聞かれたら入力します。英語だと、“Update” をクリック “Install” をクリック とにかく適当にクリックなりパスワード入力なりする感じ？
5. 言語 (Language) タブで、キーボード入力に使う IM システム (Keyboard input method system) として ibus を選択。
6. 日本語がリストにない場合、“Install / Remove Languages...” をクリック。Japanese をチェックして “Apply Changes” をクリック。パスワードを入力。
7. 一番下にある「日本語」を一番上までドラッグ

なお、日本語環境をもっと充実させたい場合は、<https://www.ubuntulinux.jp/japanese> の「方法 2」を試すとよいです（「Ubuntu 13.10 の場合」の saucy.list を trusty.list に書き換え。）ただし、これをやるとまたネットワークが重くなるので、この授業中にはやらないでください。具体的には、

```
wget -q https://www.ubuntulinux.jp/ubuntu-ja-archive-keyring.gpg -O- | sudo apt-key add -
wget -q https://www.ubuntulinux.jp/ubuntu-jp-ppa-keyring.gpg -O- | sudo apt-key add -
sudo wget https://www.ubuntulinux.jp/sources.list.d/trusty.list -O /etc/apt/sources.list.d/
sudo apt-get update
sudo apt-get upgrade
sudo apt-get install ubuntu-defaults-ja
```

を実行して、メニューの Preferences 以下の「言語サポート (Language Support)」を起動します。

## 5.2 最新ソフトウェアパッケージへのアップデート、および自動アップデートの設定

最新パッケージへのアップデートには、Software Updater を使用します。

1. メニューの System Tools 以下の Software Updater を起動します。
2. アップデートがあるけど今インストールするかと聞いてくるが、授業中にやるとネットワークが重くなるので今はやめておく。ただし、後でちゃんとやっておいて下さい。授業中はとりあえず設定変更だけ行います。ダイアログ左下の “Settings...” をクリックして下さい。
3. 出てきた Software and Updates ダイアログボックスで、“Update” タブの下の “When there are security updates:” のところがデフォルトで “Display immediately” になっているはずなので、“Download and install automatically” に変更しておくといよいでしょう。ついでに他にも、ダウンロード先が日本国内でない場合は、日本国内に変更したりするとよいかも。逆に、ダウンロードがうまく行かない場合はダウンロード先を他の国にするとよいです。必要に応じて、パスワードを入力すべし。設定したら、“Close” をクリックして “Software Sources” ダイアログボックスを閉じます。

4. あとでアップデートをインストールする時、ダウンロード先を変更してしまうとアップデート可能なパッケージ一覧がリセットされてしまうので、一旦“Check”をクリックしてパッケージ一覧を再構築してから“Install Updates”をクリックするとよいでしょう。
5. Linux カーネルがアップデートされた場合、終了すると再起動するか聞いてきます。“Restart Now”をクリックして再起動するか、あるいはもうやるのがなければ電源を落としてしまいましょう。

## 6 仮想マシンの終了

OS を仮想化ソフト内で実行していると、なんとなく普通のアプリケーションの感覚になって、つい仮想マシンを実行したまま PC の電源を落としそうになります。しかしそれは、いってみれば PC の電源を切るのに電源コードを引きぬくようなものです。可能な限りゲスト OS の側から電源を落とすようにしましょう。

電源の落とし方はフレーバーごと（つまりデスクトップ環境ごと）に違うのですが、Kubuntu や Lubuntu や Xubuntu（つまり KDE や LXDE や XFCE）の場合左下のメニューから、Ubuntu や Ubuntu GNOME の場合右上の電源ボタンからだったと思います。（GNOME の場合左上のメニューからだったかも）

なお、ソフトウェアのアップデート中に電源を落とすとまたややこしいことになるので、`apt-get` コマンドや Language Support コマンドの終了後に電源を落とすようにして下さい。