

# LPI-Japan 主催 LPICレベル1技術解説無料セミナー



LPI-Japanアカデミック認定校  
スキルブレイン株式会社 インストラクター  
河原木 忠司



- LPIC (Linux技術者認定試験)の概要
- 受験するに当たって  
-学習環境の準備-
- レベル1試験のポイント



## LPIC (Linux技術者認定試験)の概要



LPIC (Linux技術者認定試験)は、特定非営利活動法人/Linux技術者認定機関「LPI」(本部:カナダ)が実施している全世界共通の統合的かつ最高品質のLinux技術者認定資格です。

## ■3つの特徴

- GLOBAL : 世界標準資格  
世界共通基準で認定を行っており、150カ国以上の方々に受験されています。
- NEUTRAL : 中立・公正  
LPICはベンダーやディストリビューションに全く依存せず、常に中立公正な立場で、より幅広い見地からLinux技術力を評価しています。
- STANDARD : 世界最大規模  
LPICは2011年1月末には、全世界で28万人以上が受験し、9万5千人以上の認定者が生まれています。これはLinux技術者認定試験では世界最大です。

※LPI-Japan公式サイトより

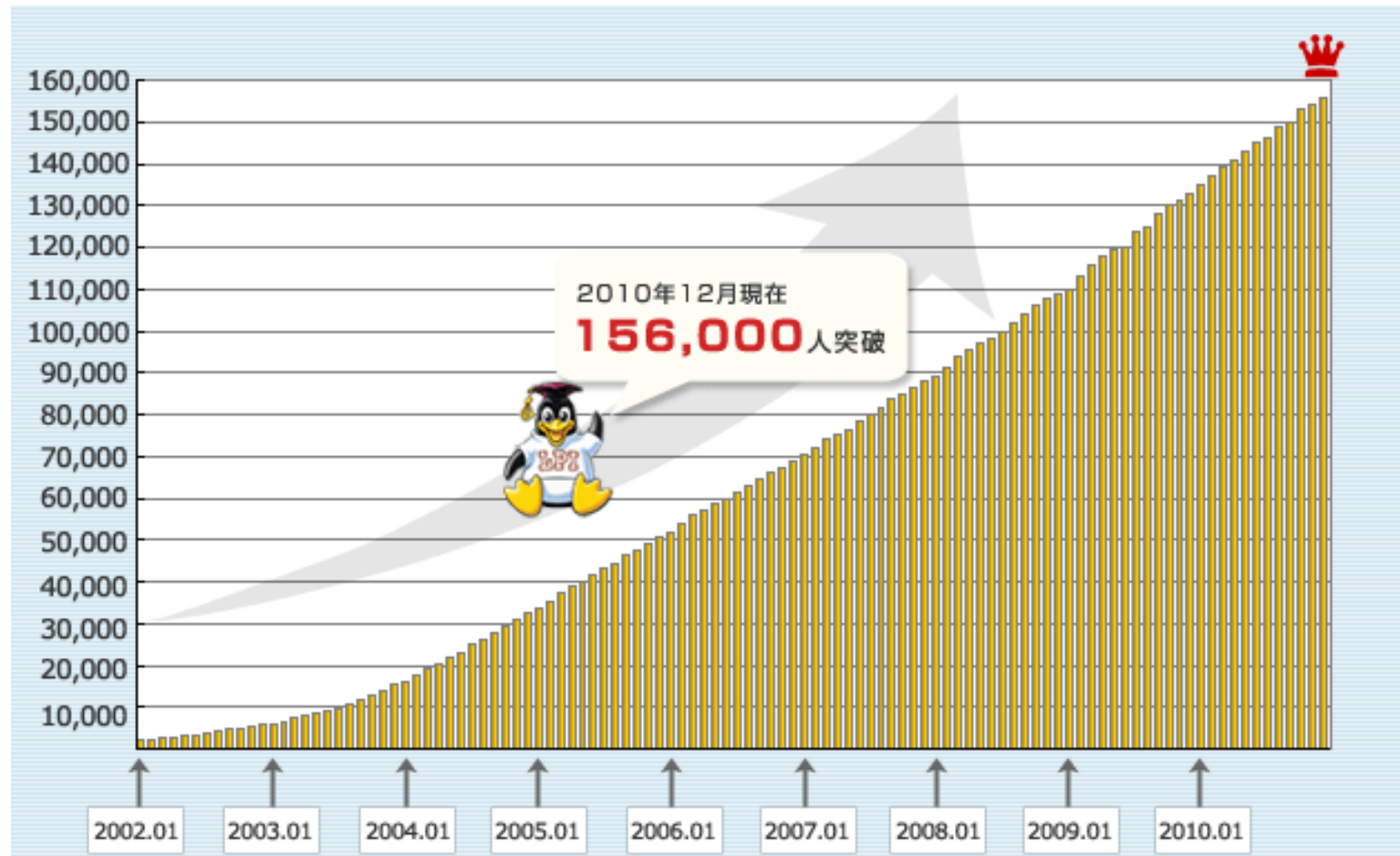




日本国内では...

Skill Brain

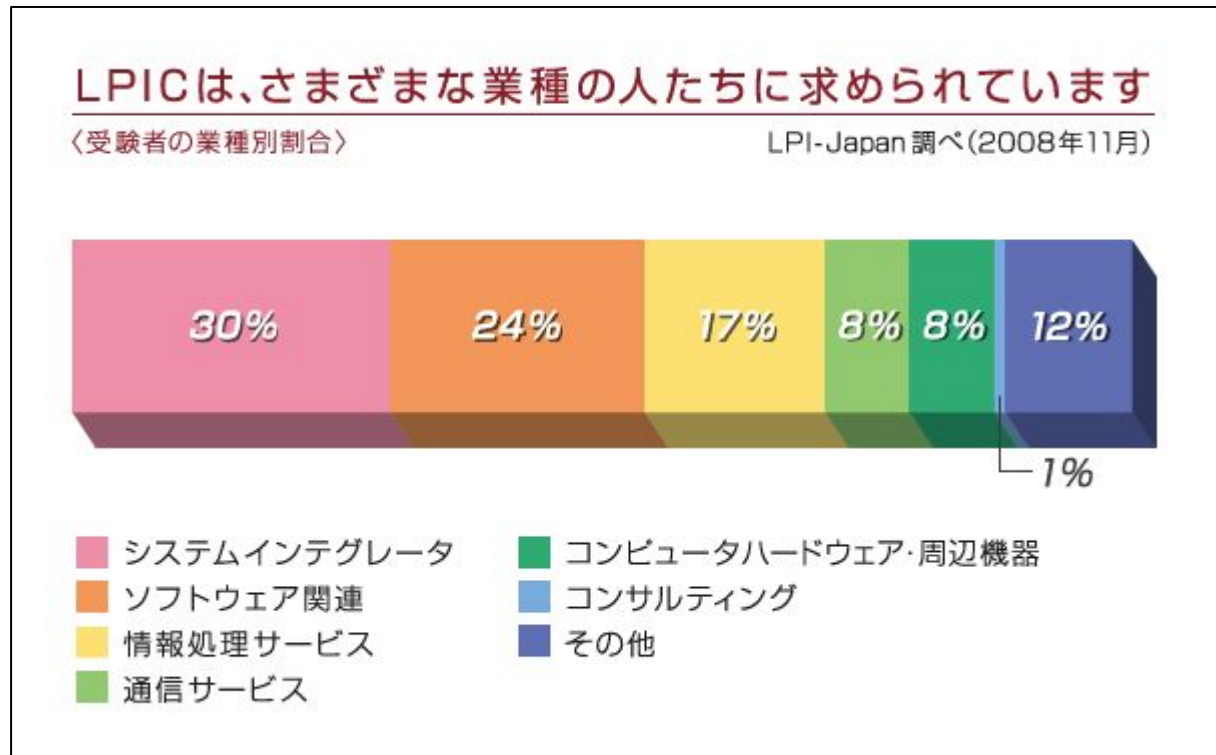
LPICは2010年12月末には日本国内での累計受験者数が15.6万人を突破



※LPI-Japan公式サイトより



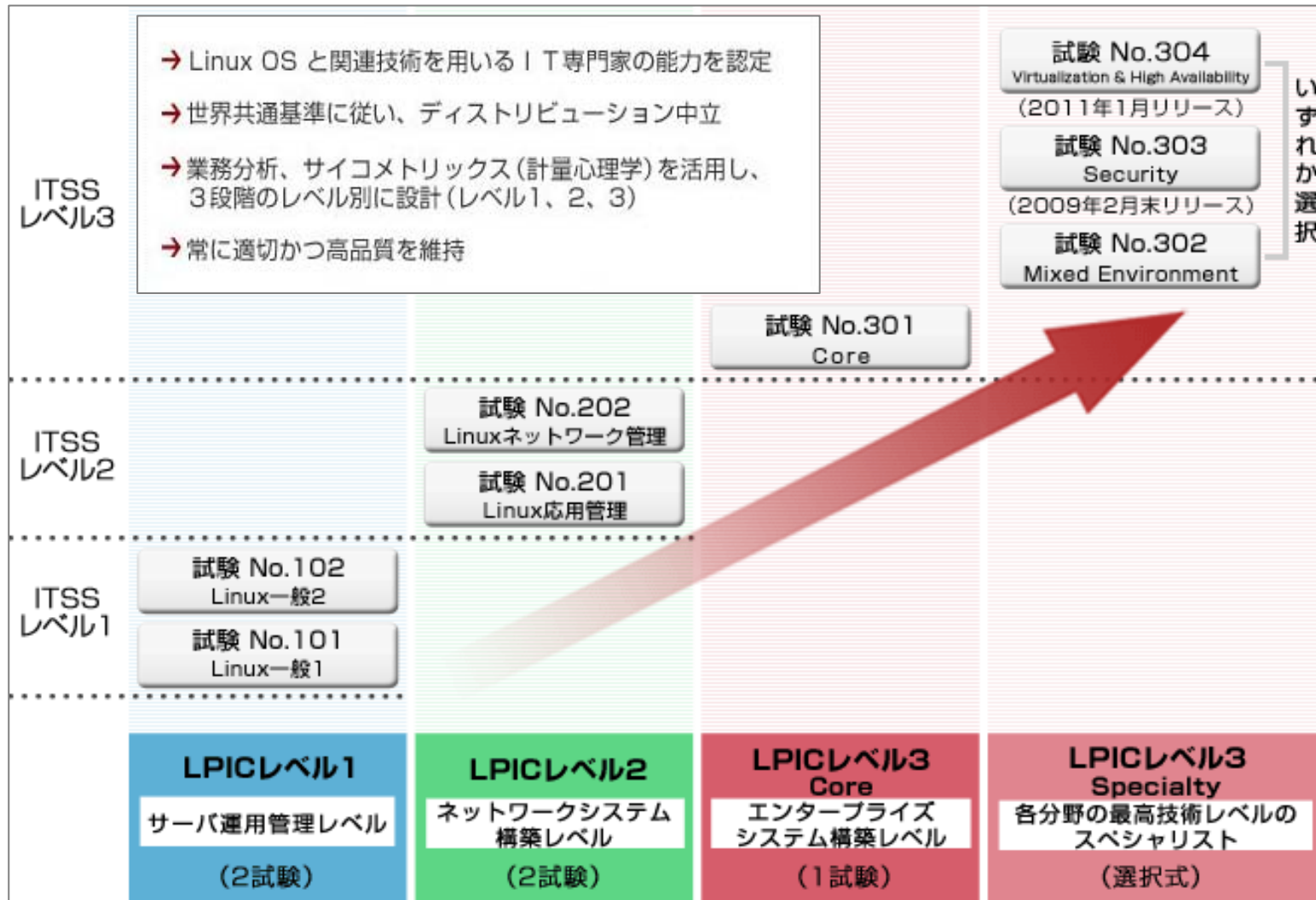
- 「営業効果の高い上位20資格」や「受験費用・講習代などの経費を補助している上位20資格」で上位にランキング（日経ソリューションビジネス調査）
- さまざまな業種の受験者層





# LPIC試験の構成

- 3段階のレベル分けがされており、レベル1試験が入り口の位置づけとなっている。





## ■レベル1

Linuxホストについて一通りの操作ができる。

- 101【Linux一般1】: 基本的なコマンドの使い方など。
- 102【Linux一般2】: システム管理的な操作など。

2科目とも合格すると、  
レベル1認定

## ■レベル2

全般的なサーバー構築・管理などの操作ができる。

## ■レベル3

特定の分野について特化した知識を身につけている。





## ■ 101試験

- 主題101: システムアーキテクチャ
- 主題102: Linuxのインストールとパッケージ管理
- 主題103: GNUとUnixのコマンド
- 主題104: デバイス、Linuxファイルシステム、ファイルシステム階層標準

## ■ 102試験

- 主題105: シェル、スクリプト、およびデータ管理
- 主題106: ユーザーインターフェイスとデスクトップ
- 主題107: 管理業務
- 主題108: 重要なシステムサービス
- 主題109: ネットワークの基礎
- 主題110: セキュリティ

※公式サイトURL

<http://www.lpi.or.jp/lpic1/range/>



## 受験するに当たって -学習環境の準備-

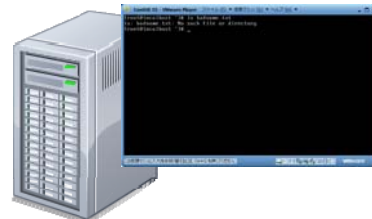


## ■想定される学習環境

- 実機環境



- 仮想環境



- テキストでのみ学習

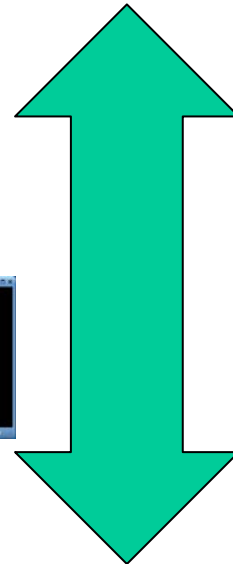


手間  
かかる

理解  
深い

簡単

浅い





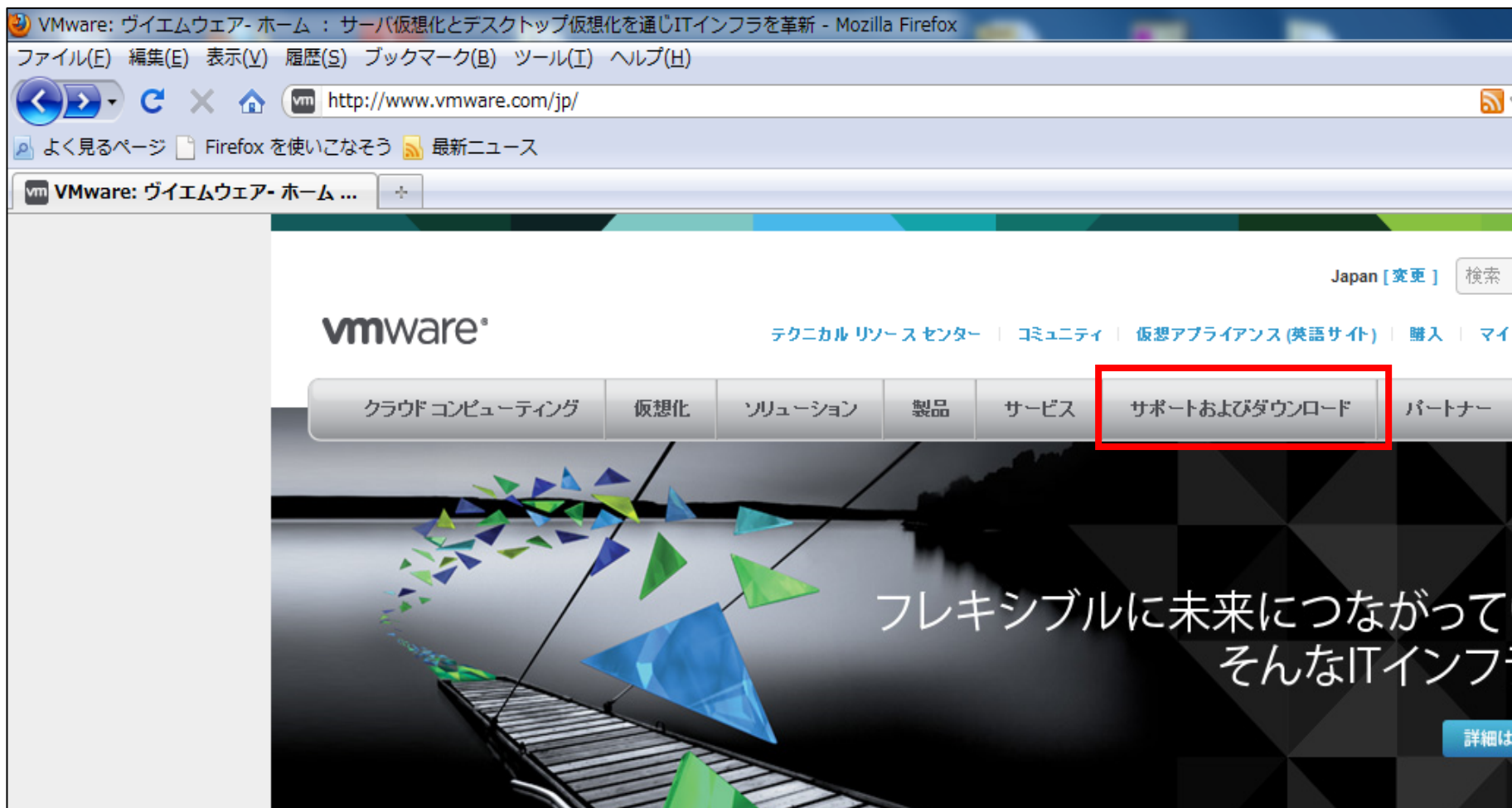
## ■仮想環境導入のメリット

- 既存のOS環境をそのまま利用して、Linux環境を構成することができる。
- 複数の環境を容易に利用することができる。
- 環境を破壊するような操作を試みることもできる。
- テキストの付録などを利用し、手軽に環境を構築することができる。



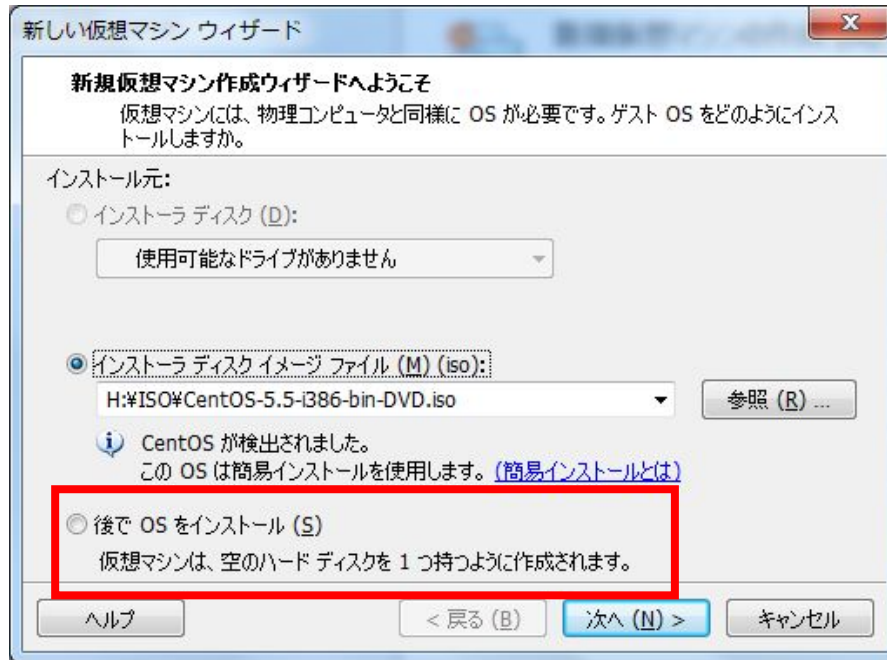
## ■ VMWare

<http://www.vmware.com/jp/>





- [新規仮想マシンの作成]  
一からインストールを行う場合に利用。
- [仮想マシンを開く]  
既存の仮想マシン環境が存在する場合に利用。  
※テキストによっては付録で仮想マシン環境を用意してくれているものもある。



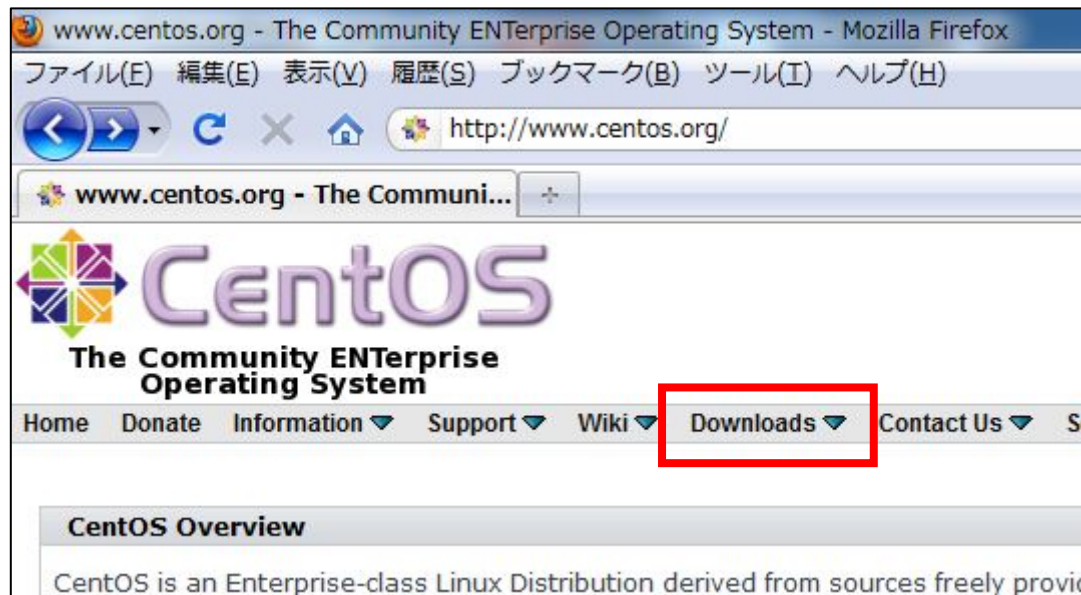
- [後でOSをインストール]を選択  
[インストーライメージファイル]を選択し、簡易インストール機能を利用することもできるが、自動的にインストール処理が実行されるため、インストール構成をカスタマイズできない。



- 各ディストリビューションのインストールイメージ(iso)ファイルを手入。
- インストールイメージファイルをマウントし、仮想ホストを起動。
- インストーラが起動するので、セットアップを実行。

## CentOS公式サイト

<http://www.centos.org/>

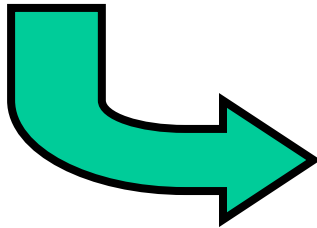






## ■Linuxとは？

- 狭義(本来)の意味ではカーネル
- 広義の意味ではOS全般



- OSとしての配布形態
- 様々な組織等が様々なディストリビューションを提供している。
  - RedHat系ディストリビューション
    - Red Hat Enterprise Linux
    - Fedora
    - CentOS
  - Debian系ディストリビューション
    - Debian Linux
    - Ubuntu Linux

など



作成したホストのイメージを選択し、  
[仮想マシン設定の編集]をクリック

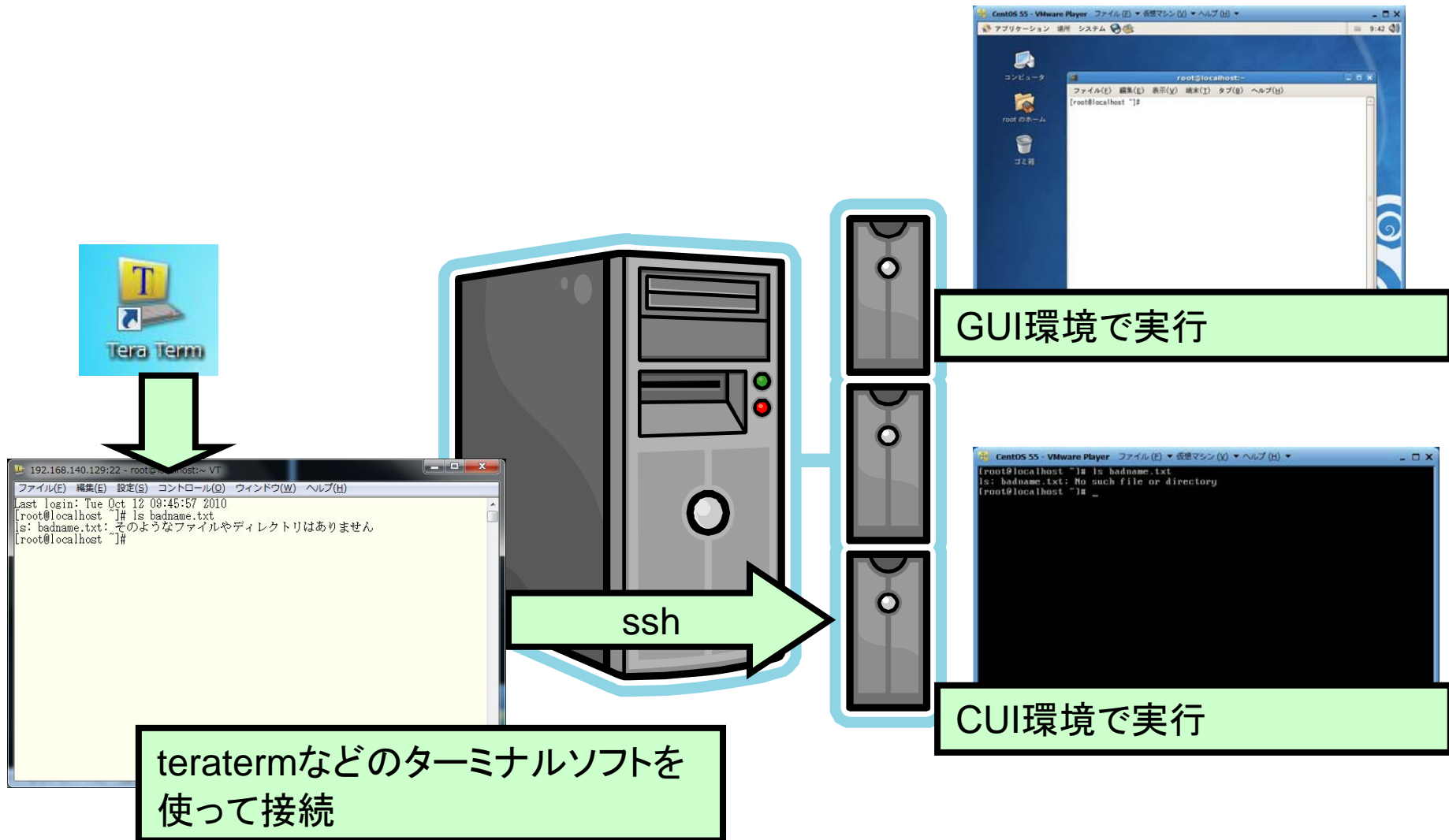
仮想マシン設定

デバイス	概要
メモリ	512 MB
プロセッサ	1
ハードディスク (SCSI)	20 GB
CD/DVD (IDE)	H:\ISO\CentOS-5.5-i386-bin-DVD...
フロッピー	自動検出
ネットワーク アダプタ	NAT
USB コントローラ	存在します
サウンド カード	自動検出
プリンタ	存在します
ディスプレイ	自動検出

仮想マシンの再生 (L)  
仮想マシン設定の編集 (D)

状態: パワーオフ  
OS: Red Hat Enterprise Linux 5  
バージョン: Workstation 6.5-7.x 仮想マシン  
RAM: 512 MB

ホストの設定画面で、CD/DVDを選択し、  
[ISOイメージファイルを使用する]をチェッ  
クし、入手したISOイメージファイルを選択





# LPICレベル1 試験のポイント



- 主題101：システムアーキテクチャ
- 主題102：Linuxのインストールとパッケージ管理
- 主題103：GNU / UNIXコマンド
- 主題104：デバイス、Linuxファイルシステム、ファイルシステム階層標準



- 101.1 ハードウェア設定の決定と構成
- 101.2 システムのブート
- 101.3 ランレベルの変更とシステムのシャットダウンまたはリブート



## ■ランレベルとは？

Linuxシステムの稼働状態を表す。

- `init 3`                      ランレベルの変更  
  `telinit 5`
- `runlevel`                    現在稼働しているランレベルの表示
- `/etc/inittab`                initの設定ファイル・デフォルトのランレベルの設定

## ■CentOSのランレベル

ランレベル	説明
0	システムの停止
1	シングルユーザーモード
3	マルチユーザーモード(CUI)
5	マルチユーザーモード(GUI)
6	システムの再起動



- 102.1 ハードディスクのレイアウト設計
- 102.2 ブートマネージャのインストール
- 102.3 共有ライブラリを管理する
- 102.4 Debianパッケージ管理を使用する
- 102.5 RPMおよびYUMパッケージ管理を使用する





- Linuxディストリビューションでは、アプリケーションをパッケージという単位で、管理している。
- それぞれのパッケージ管理形式で、インターネット上のリポジトリからパッケージを入手できる機能が用意されている。

パッケージ管理形式	コマンド	ファイル形式	設定ファイル
RPM形式	rpm, yum	～.rpm	/etc/yum.conf /etc/yum.repos.d/
Debian形式	dpkg, apt-get	～.deb	/etc/apt/sources.list



- 102試験の範囲を学習する際に、必要なパッケージをインストール。

yum groupinstall "X Window System"

yum groupinstall "GNOME Desktop Environment"

yum groupinstall "MySQL Database"

yum install ntp

} 主題106  
X環境の学習で  
利用

————— 主題105  
SQLコマンドの学習で利用

————— 主題108  
NTPの学習で利用



- 103.1 コマンドラインで操作する
- 103.2 フィルタを使ってテキストストリームを処理する
- 103.3 基本的なファイル管理を行う
- 103.4 ストリーム、パイプ、リダイレクトを使う
- 103.5 プロセスを生成、監視、終了する
- 103.6 プロセスの実行優先度を変更する
- 103.7 正規表現を使用してテキストファイルを検索する
- 103.8 viを使って基本的なファイル編集を行う



[user@localhost ~]\$ ls -l /etc

プロンプト

コマンド

オプション

引数

プロンプト	ログインしているユーザー名やホスト名、カレントとなっているディレクトリを表示
コマンド	実行するコマンドを指定
オプション	必要に応じて、オプションを設定し、コマンドの挙動を変えることができる
引数	必要に応じて、引数を設定し、コマンド実行時に値を渡すことができる。



## ■ ディレクトリ

- ファイルを格納する「入れ物」。
- Windowsの「フォルダ」



## ■ ディレクトリ操作に関する主なコマンド

ls	ディレクトリの一覧を表示
pwd	カレントディレクトリを表示
cd	カレントディレクトリを変更
mkdir	ディレクトリを作成

例) /etcディレクトリの内容を表示

```
[root@localhost ~]# ls /etc
```



## ■カレントディレクトリ

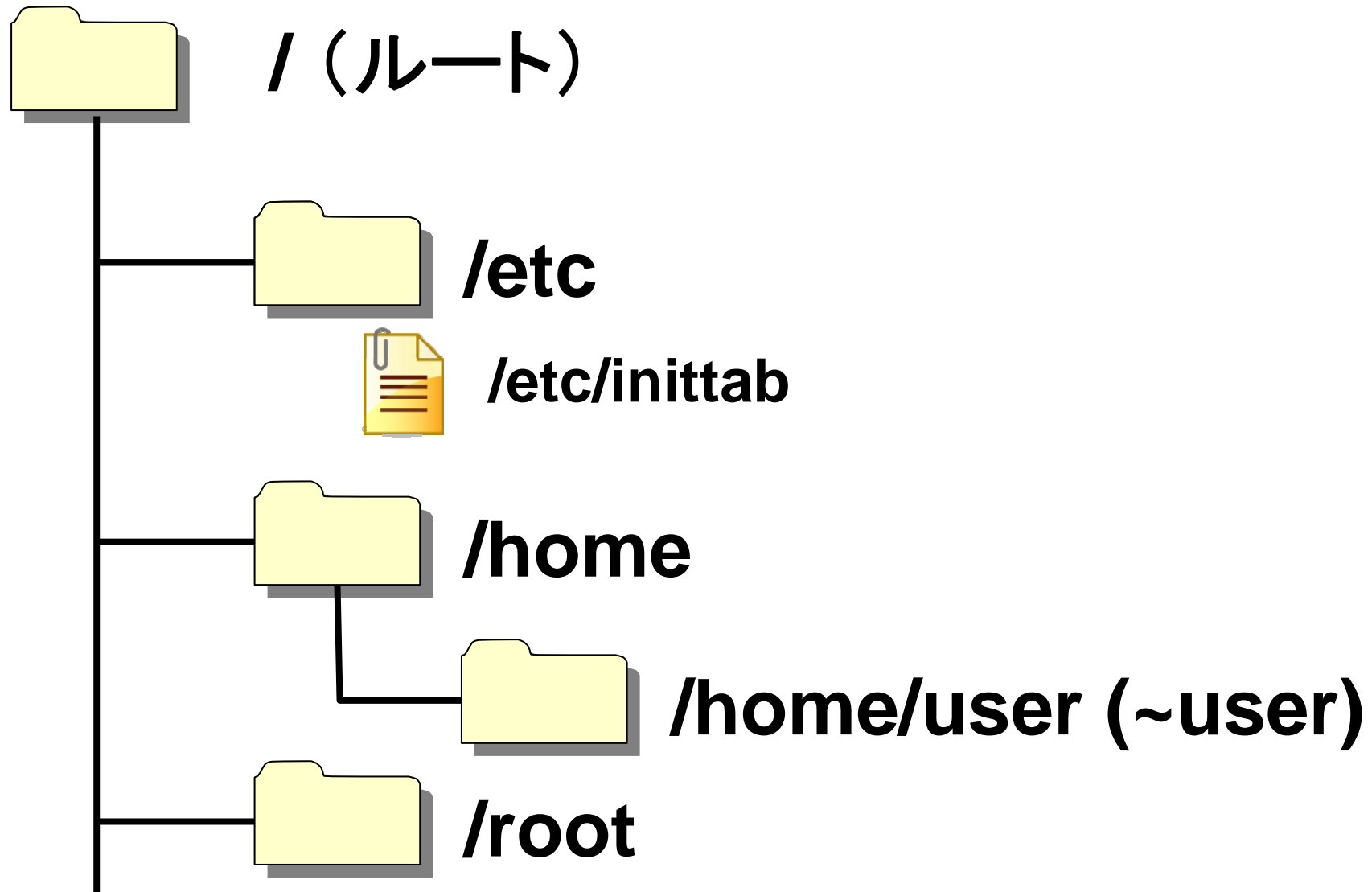
- ユーザーの作業領域となるディレクトリ
- ディレクトリやファイルの指定の基準とすることができる

例)

カレントディレクトリが/etcの場合、/etcディレクトリの内容を表示

```
[root@localhost etc]# ls /etc  
[root@localhost etc]# ls
```

カレントディレクトリが/etcであるため、ディレクトリの指定を省略することができる。





.	カレントディレクトリを指定
..	1つ上のディレクトリを指定
~	ホームディレクトリを指定
~kawaragi	ユーザーkawaragiのホームディレクトリを指定

例)

rootユーザーでログインしている場合、rootユーザーのホームディレクトリ(/root)の内容を表示

```
[root@localhost etc]# ls /root
[root@localhost etc]# ls ~
[root@localhost etc]# ls ~root
```



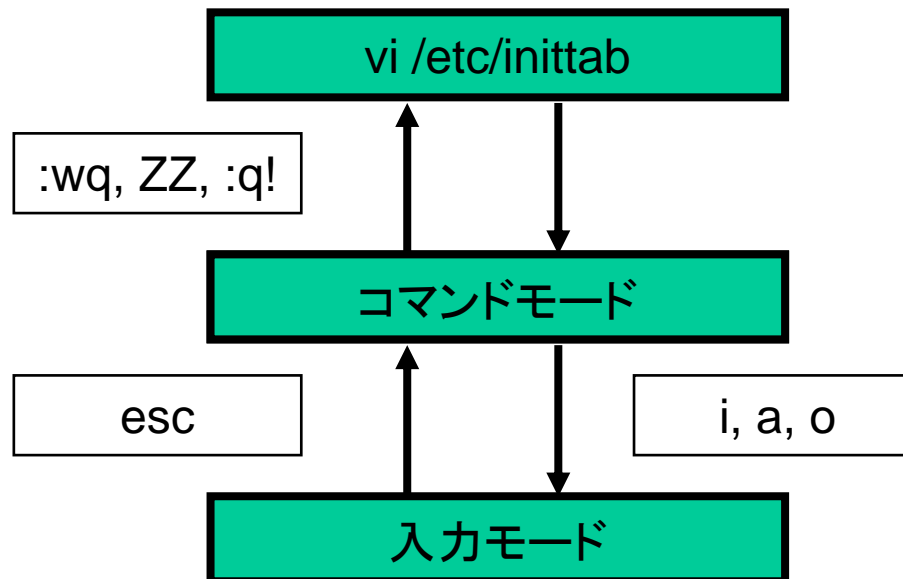


## ■ cat, head, tail

ファイルの内容を参照

## ■ vi

viエディタでファイルを編集





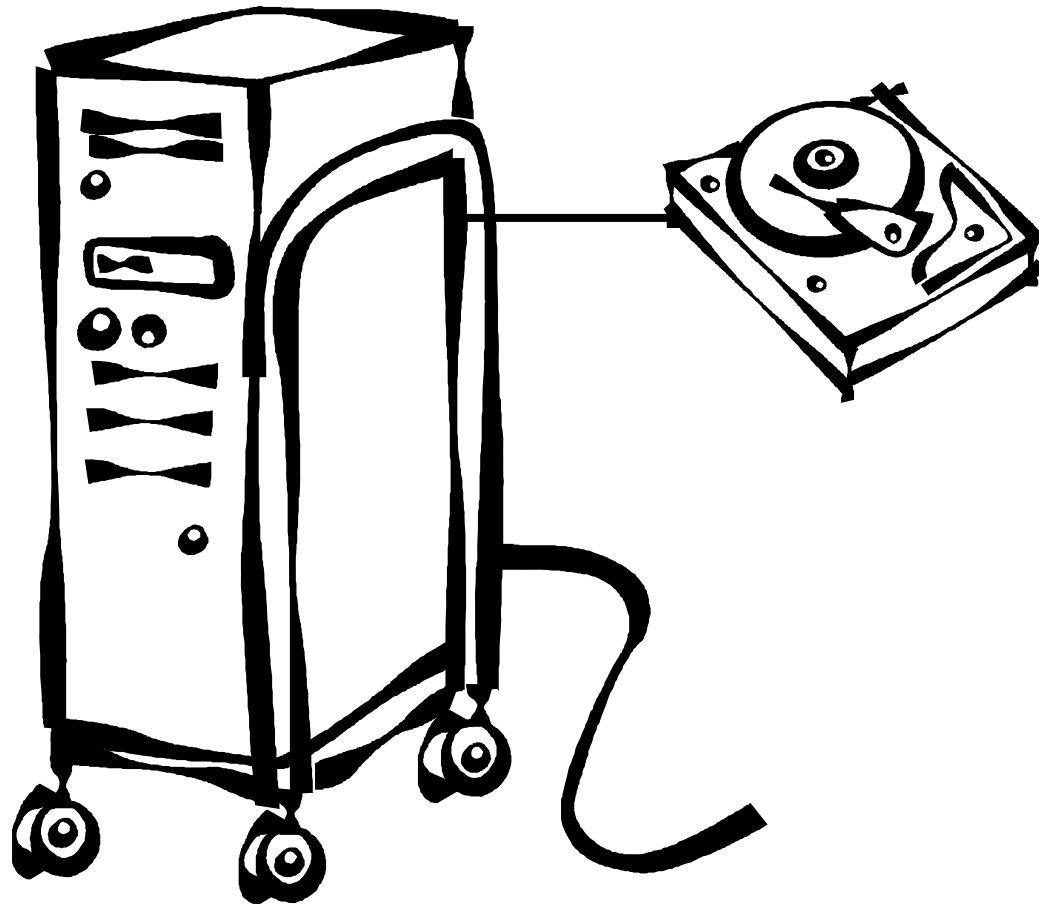
- 記号などの特殊文字を利用し、文字列パターンを表現する手法
- grepやsedなどと併用し、条件に合致した、もしくは合致しない行だけで抽出して、表示できる。
- 例>  

```
[root@localhost ~]# grep -v "^#" /etc/httpd/conf/httpd.conf | grep -v "^$"
```

記号	意味
.	任意の一文字
*	直前の文字の0回以上の繰り返し
^	行頭
\$	行末

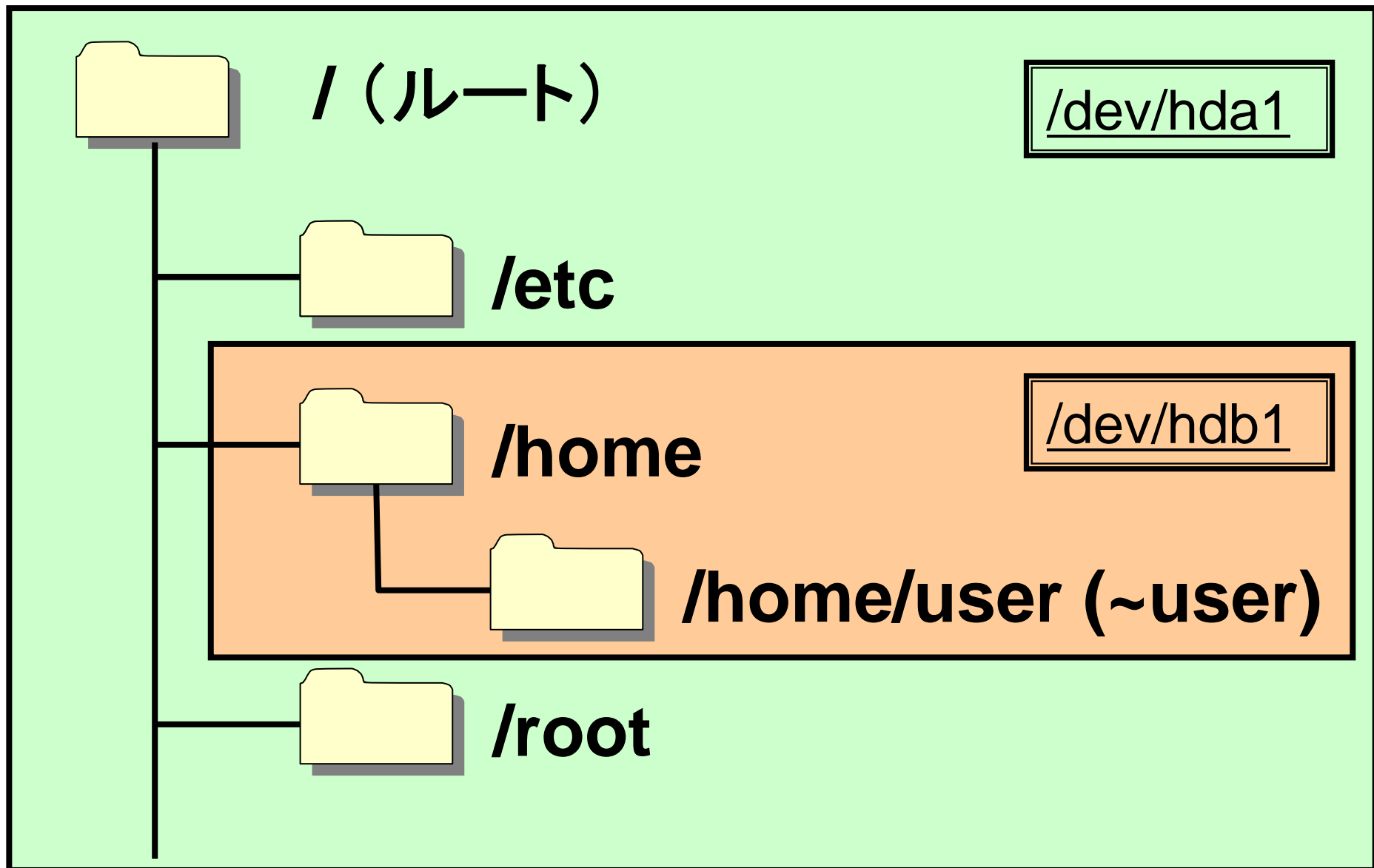


- 104.1 パーティションとファイルシステムの作成
- 104.2 ファイルシステムの整合性を保持する
- 104.3 ファイルシステムのマウントとアンマウントをコントロールする
- 104.4 ディスククォータを管理する
- 104.5 ファイルのパーミッションと所有者を管理する
- 104.6 ハードリンクとシンボリックリンクを作成・変更する
- 104.7 システムファイルを見つける、適切な位置にファイルを配置する



/dev/hda

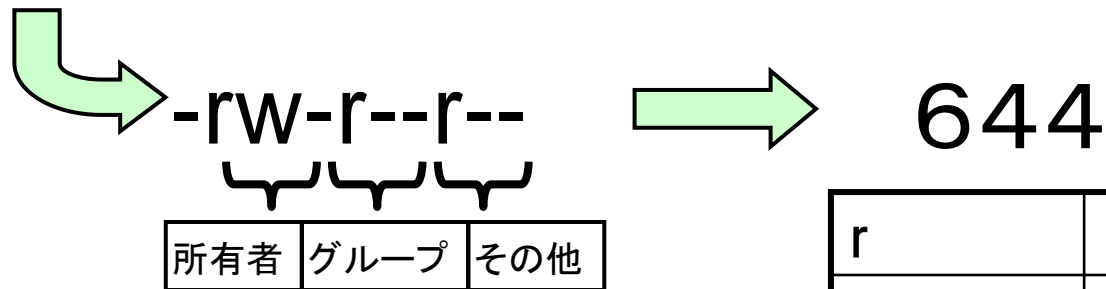
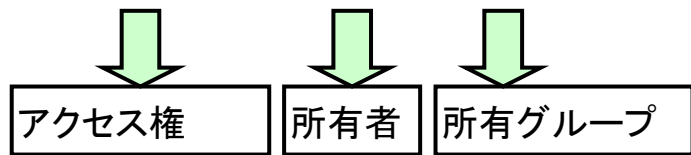
- hd : IDE形式のディスク
  - hda : プライマリマスターに接続
  - hdb : プライマリスレーブに接続
  - hdc : セカンダリマスターに接続
  - hdd : セカンダリスレーブに接続
- sd : SCSI, USBなどの形式のディスク





```
[root@localhost ~]# ls -l sample.txt
```

```
-rw-r--r-- 1 root root 26 12月 9 22:52 sample.txt
```



r	4	読み取り
w	2	書き込み
x	1	実行



- LinuxやほかのUNIX系OSにおいて、ディレクトリ構造とそれぞれのディレクトリの役割を示した標準仕様。

ディレクトリ	役割
/boot	ブートローダ関連のファイル
/dev	デバイスファイル
/etc	システム全般の設定ファイル
/home	ユーザーのホームディレクトリ
/root	rootユーザーのホームディレクトリ



- 主題105：シェル、スクリプト、およびデータ管理
- 主題106：ユーザーインターフェースとデスクトップ
- 主題107：管理業務
- 主題108：重要なシステムサービス
- 主題109：ネットワークの基礎
- 主題110：セキュリティ





- 105.1 シェル環境のカスタマイズと使用
- 105.2 簡単なスクリプトをカスタマイズまたは作成する
- 105.3 SQLデータ管理



- テキストファイルの中にコマンドを記述しておき、これをまとめて実行することができる。

```
[root@localhost ~]# cat cp_new.sh
```

```
#!/bin/bash  
if [ $1 -nt $2 ]  
then  
    cp $1 $2  
else  
    cp $2 $1  
fi
```

```
[root@localhost ~]# chmod a+x cp_new.sh
```

```
[root@localhost ~]# ./cp_new.sh sample1.txt sample2.txt
```

シェルスクリプトの先頭行にはシェルの実行パスを指定

\$1, \$2はシェルスクリプト実行時の引数。  
新しいファイルがどちらか比較をして、新しい方の内容をコピーする

スクリプトには実行権限を与える

カレントディレクトリに存在するスクリプトを実行する場合、./スクリプト名と記述



## ■ 変数

- 変数名=値  
\$変数名で値を参照することができる
- 特殊な変数
  - \$1, \$2 ... : 引数
  - \$# : 引数の数

## ■ testコマンド、[]

条件を設定し、true、falseを返す

- 数値による条件
  - 値1 -eq 値2: 値1と値2が等しい
  - 値1 -ne 値2: 値1と値2が等しくない
  - 値1 -ge 値2: 値1が値2以上
  - 値1 -gt 値2: 値1が値2より大きい
  - 値1 -le 値2: 値1が値2以下
  - 値1 -lt 値2: 値1が値2より小さい
- ファイル属性による条件
  - -f ファイル: ファイルが存在する
  - ファイル1 -nt ファイル2: ファイル1がファイル2より新しい



## ■ if文: 条件分岐

if 条件式

then

    true(真)の場合の処理→条件に合致した場合の処理

else

    false(偽)の場合の処理→条件に合致しなかった場合の処理

fi

## ■ case文: 条件分岐(複数の処理)

case 変数名 in

    値1) 処理1 ;;

    値2) 処理2 ;;

    \*) 上記の値以外の場合の処理 ;;

esac



■ for文: 値リストを利用した繰り返し処理

```
for 変数名 in 値リスト
```

```
do
```

```
    値リストの値に対する処理
```

```
done
```

■ while文: 条件を指定した繰り返し処理

```
while 条件式
```

```
do
```

```
    条件に合致している限り繰り返す処理
```

```
done
```



```
[root@localhost ~]# /etc/init.d/mysqld start
MySQL を起動中: [ OK ]
[root@localhost ~]# mysql -u root
Welcome to the MySQL monitor. Commands end with ; or ¥g.
Your MySQL connection id is 2
Server version: 5.0.77 Source distribution
```

mysqldの起動

mysqldに接続

Type 'help;' or '¥h' for help. Type '¥c' to clear the buffer.

```
mysql> show databases;
```

```
+-----+
| Database |
+-----+
| information_schema |
| mysql |
| test |
+-----+
```

データベース一覧を表示

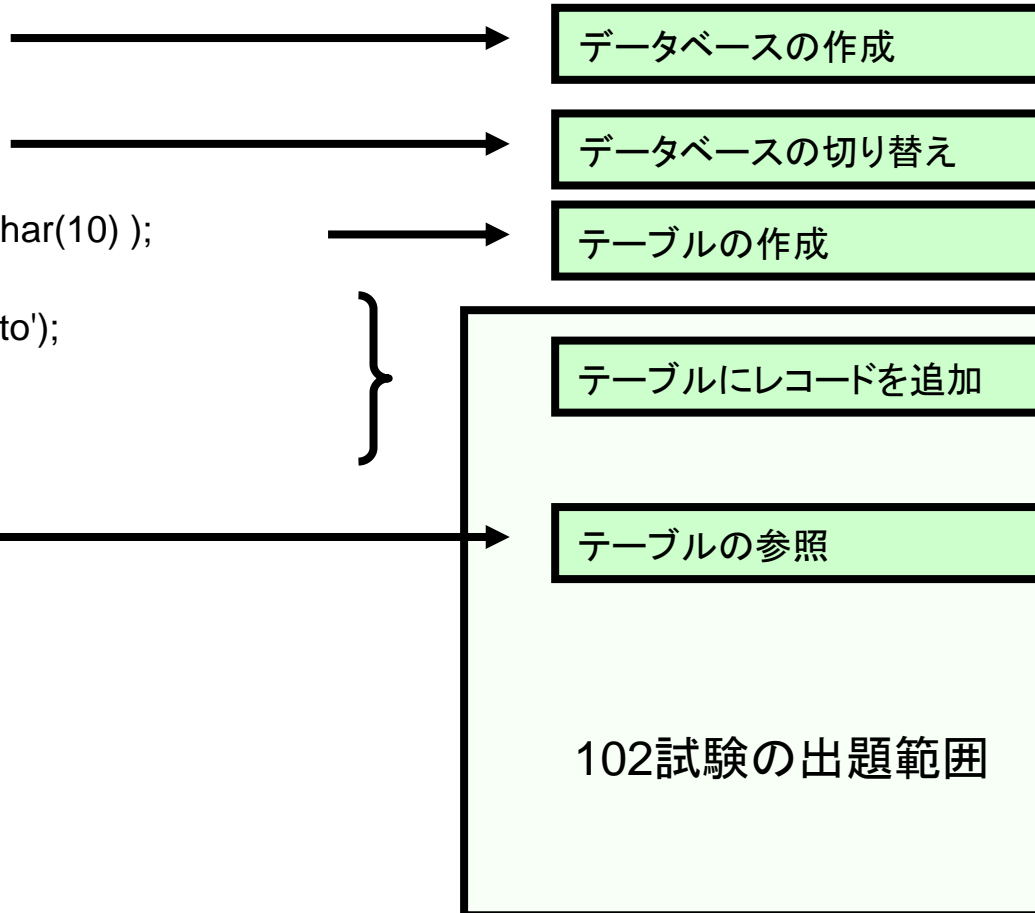
```
3 rows in set (0.00 sec)
```

```
mysql> quit
Bye
[root@localhost ~]#
```

mysqldへの接続を切断



```
[root@localhost ~]# mysql -u root
mysql> create database testDB;
Query OK, 1 row affected (0.00 sec)
mysql> use testDB;
Database changed
mysql> create table staff( id int, name varchar(10) );
Query OK, 0 rows affected (0.00 sec)
mysql> insert into staff values (1, 'Nakamoto');
Query OK, 1 row affected (0.00 sec)
mysql> insert into staff values (2, 'Takagi');
Query OK, 1 row affected (0.00 sec)
mysql> select * from staff;
+-----+-----+
| id  | name  |
+-----+-----+
|  1  | Nakamoto |
|  2  | Takagi  |
+-----+-----+
2 rows in set (0.00 sec)
```





■ SQL文の一部で、テーブル内のレコードの追加・検索・更新・削除などを行なう際に使用する。

## ■ 主なDML文

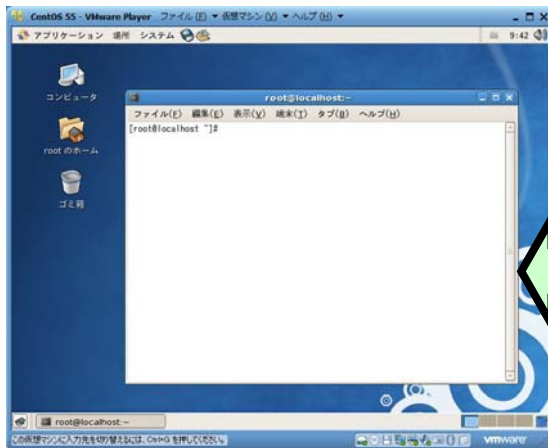
- SELECT 列名 FROM テーブル名 WHERE 列名 = 値;
- INSERT INTO テーブル名 (列名1, ...) VALUES (値1, ...);
- UPDATE テーブル名 SET 列名 = 値 WHERE 列名 = 値;
- DELETE FROM テーブル名 WHERE 列名 = 値;



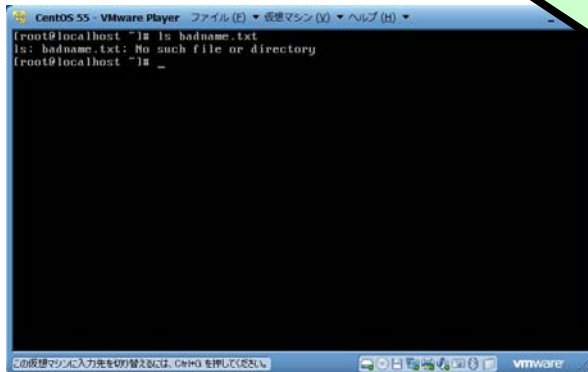


## 主題106：ユーザーインターフェースとデスクトップ Skill Brain

- 106.1 X11のインストールと設定
- 106.2 ディスプレイマネージャの設定
- 106.3 アクセシビリティ



[root@localhost ~]# vi /etc/inittab  
id:5:initdefault: →デフォルトのランレベル



[root@localhost ~]# startx

X環境の設定ファイル : /etc/X11/xorg.conf



- GUI環境を実現するために必要なディスプレイマネージャ(DM)、ウィンドウマネージャなどをまとめたソフトウェア
- ディスプレイマネージャ
  - XDM : デフォルトのディスプレイマネージャ
  - GDM : GNOMEで利用
  - KDM : KDEで利用



- 107.1 ユーザアカウント、グループアカウント、および関連するシステムファイルを管理する
- 107.2 ジョブスケジューリングによるシステム管理業務の自動化
- 107.3 ローカライゼーションと国際化



- /etc/passwd : ユーザー情報を保存

```
test01:x:500:600::/home/test01:/bin/bash
```

ユーザー名

- /etc/shadow : パスワード情報を保存

```
test01:$1$HTUBwjdV$w7YdIKDg/3KRz73D2910F/:14952:0:99999:7:::
```

GID

- /etc/group : グループ情報を保存

```
sales:x:600:
```



## ■ ジョブのスケジューリング

- cron: 繰り返し実行。crontabコマンドで設定
- at: 1回だけ実行。atコマンドで設定

```
[root@localhost ~]# crontab -e
```

crontabエントリにジョブを追加する。

```
no crontab for root - using an empty one
```

```
crontab: installing new crontab
```

```
[root@localhost ~]# crontab -l
```

追加したジョブの内容を確認する。

```
0 * * * * /root/sample.sh
```



## ■分 時 日 月 曜日 実行する処理

分	0～59で指定
時	0～23で指定
日	1～31で指定
月	1～12で指定、もしくはjanなど文字列で指定
曜日	0～7で指定。0と7が日曜日、1が月曜日・・・

## ■ワイルドカード、複数の値の指定

- 0 \* \* \* \* → 毎時0分に処理を実行(「分」以外の部分はすべてOK)
- 0 \*/2 \* \* \* → 2時間おきに処理を実行
- 0 22 \* \* 2,4 → 火曜日と木曜日の午後10時に処理を実行

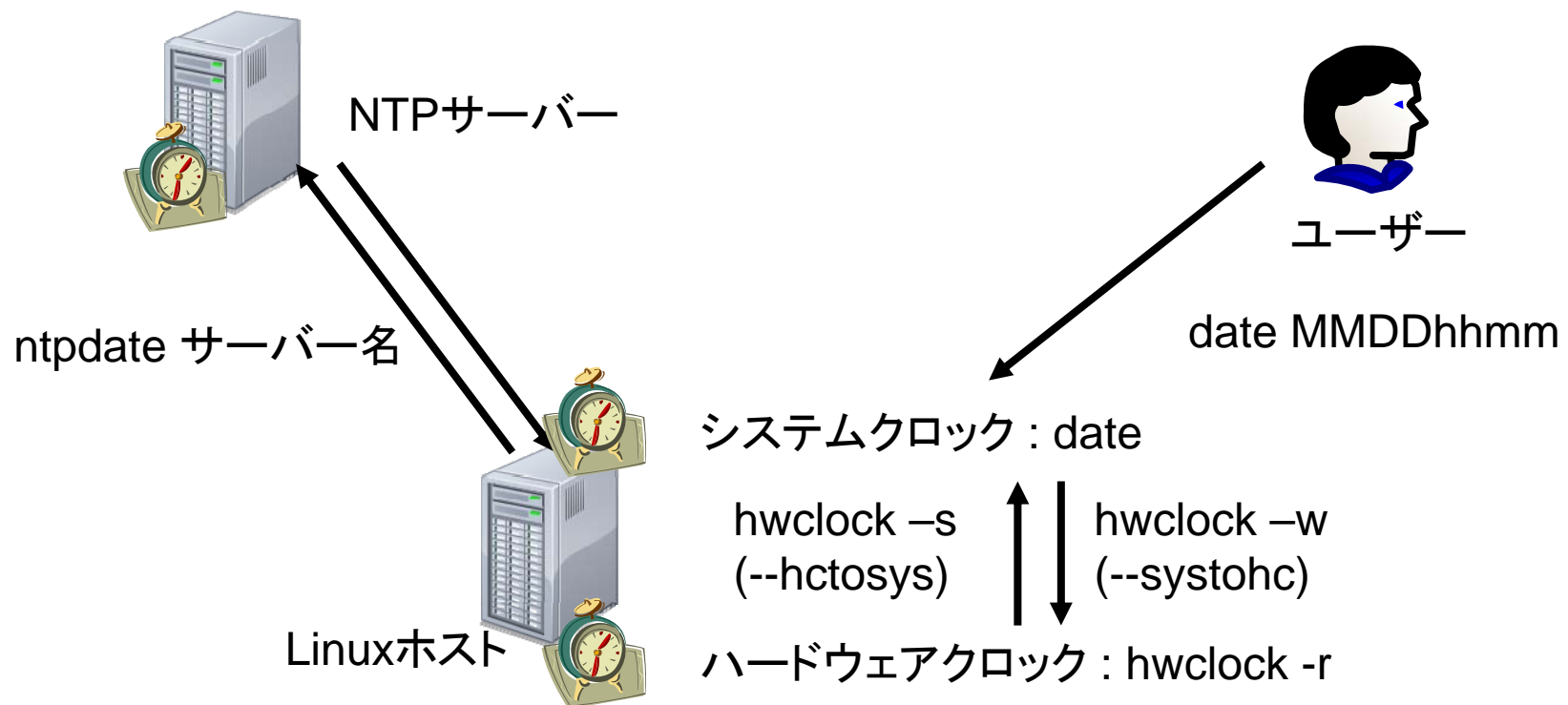


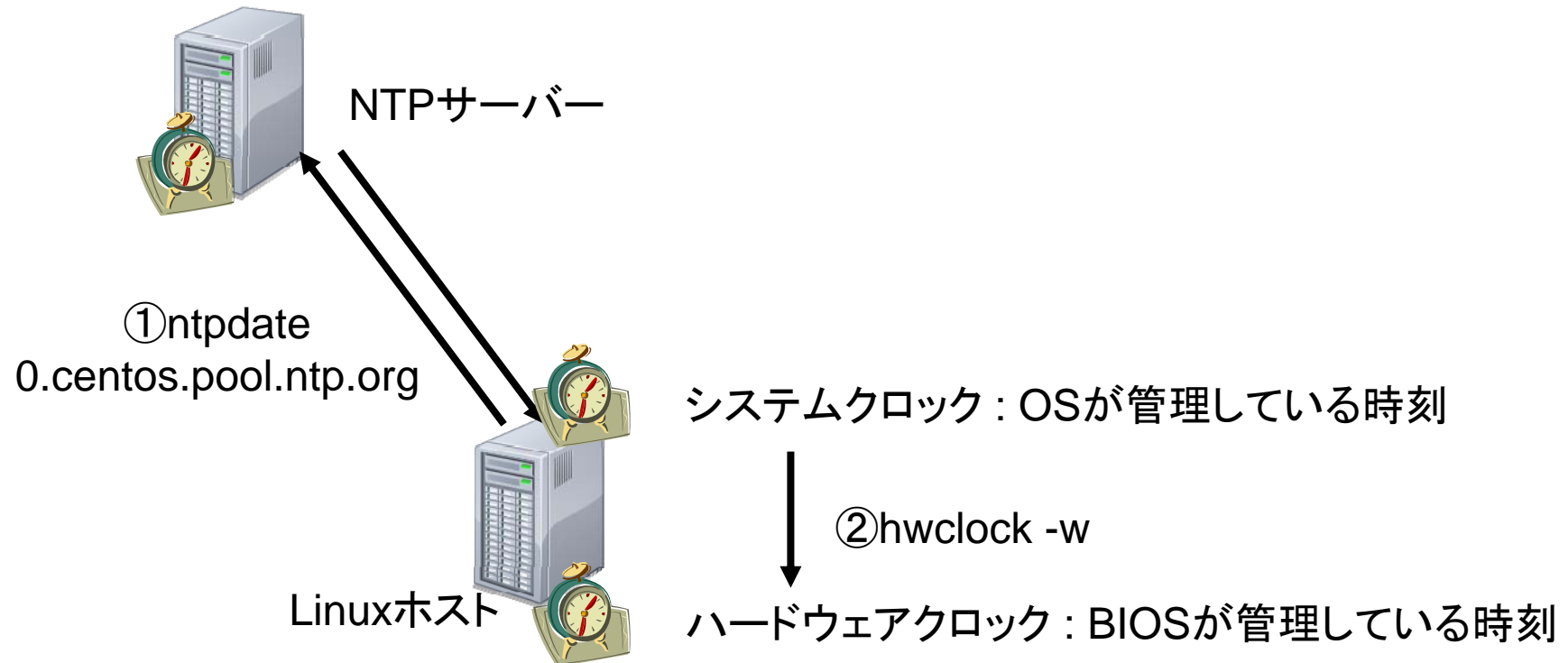
- 108.1 システム時刻を維持する
- 108.2 システムのログ
- 108.3 メール転送エージェント(MTA)の基本
- 108.4 プリンターと印刷を管理する





- `date`: システムクロックの表示・設定
- `hwclock`: ハードウェアクロックの表示・設定
- `ntpdate`: NTPサーバーとシステムクロックの同期
- `ntpd`: NTPサーバーサービス。設定ファイルは`/etc/ntp.conf`



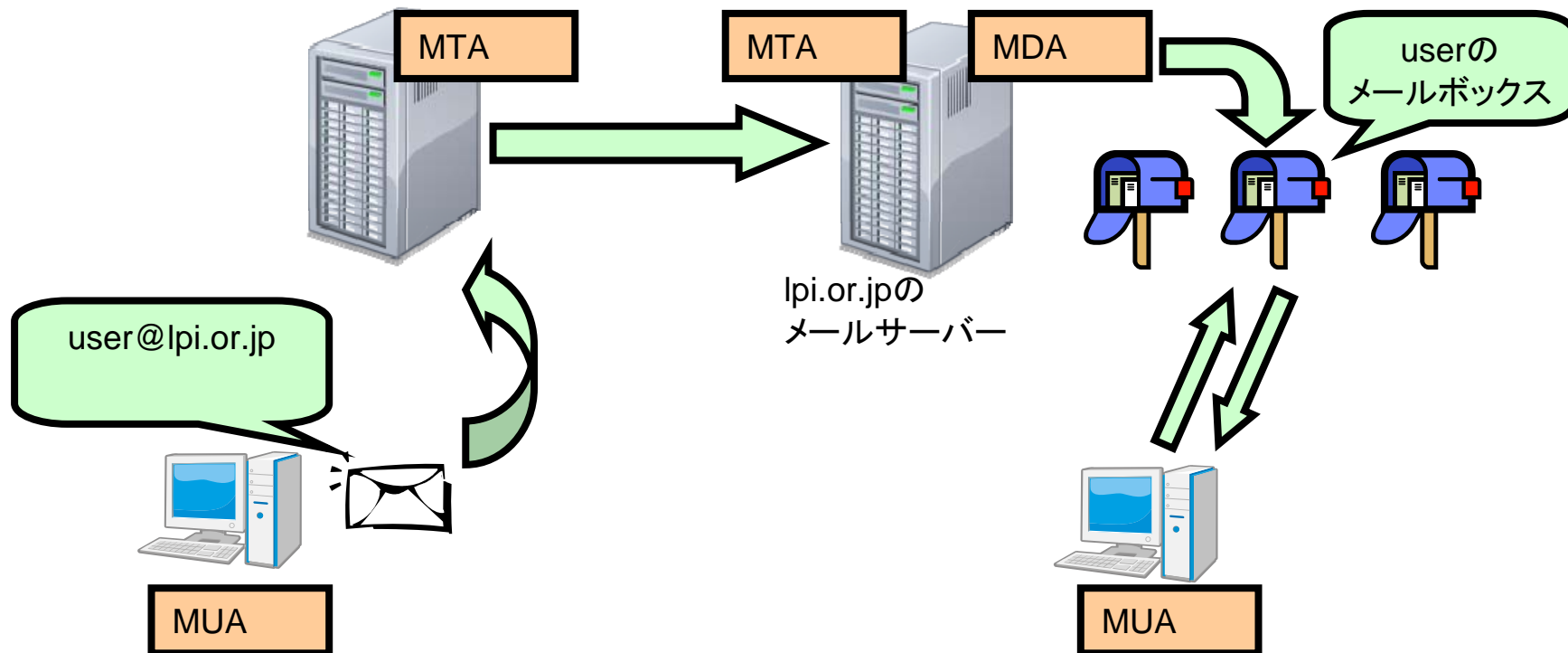


※pool.ntp.org  
公開NTPサーバーの情報を共有し、ラウンドロビンでアクセス  
させるプロジェクト



■「一般的に使用可能なMTAプログラム (postfix、sendmail、qmail、exim) について知っている (設定は含まない)」

- MTA (Mail Transfer Agent) : メールの転送
- MDA (Mail Delivery Agent) : メールの配信
- MUA (Mail User Agent) : メールクライアント





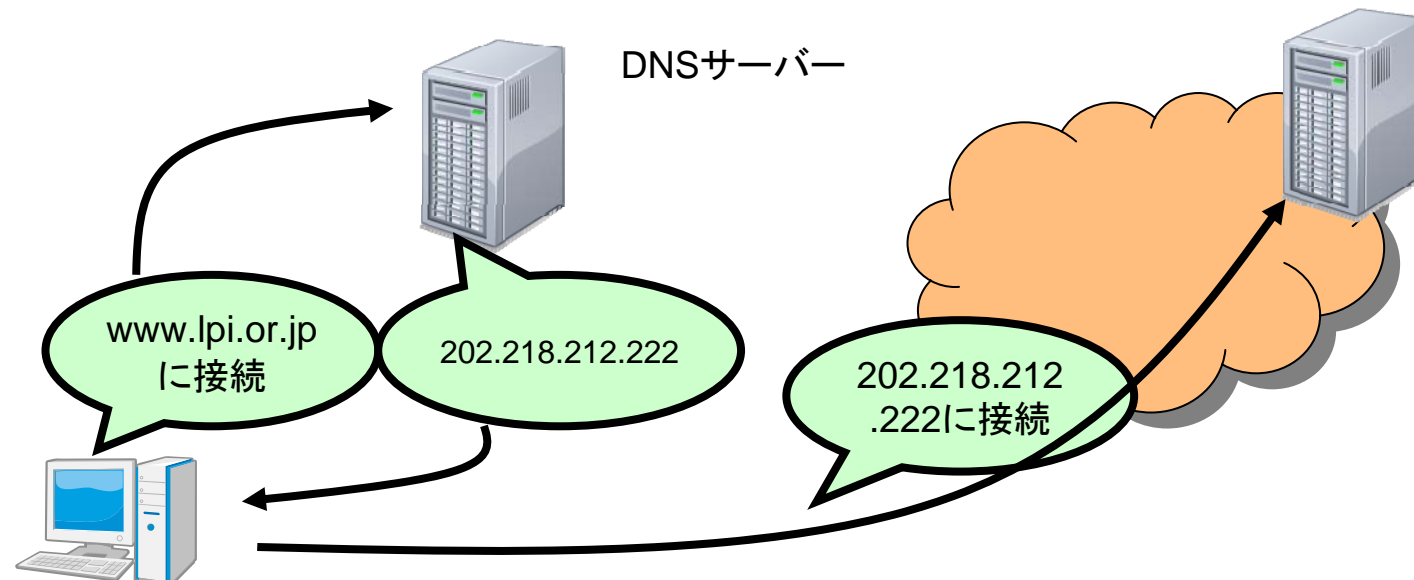
- 109.1 インターネットプロトコルの基礎
- 109.2 基本的なネットワーク構成
- 109.3 基本的なネットワークの問題解決
- 109.4 クライアント側のDNS設定



- ターミナルソフトの利用やNTPサーバとの同期の際、IPアドレス、ホスト名を指定



- TCP/IPを利用した通信では、IPアドレスを接続先・接続元のホスト情報として利用。
- コマンドの実行時など、ホスト名で指定することもできる。その場合、名前解決という処理が行われ、IPアドレスが取得される。





設定項目	コマンド名、ファイル名
IPアドレス、サブネットマスク	ifconfig
デフォルトゲートウェイアドレス	route
DNSサーバーアドレス	/etc/resolv.conf

```
[root@localhost ~]# ifconfig eth0 192.168.120.200 netmask 255.255.255.0
```

```
[root@localhost ~]# route add default gw 192.168.120.2
```

```
[root@localhost ~]# cat /etc/resolv.conf
```

```
search localdomain
```

```
nameserver 192.168.120.2
```

このほか、出題範囲に含まれる設定ファイル

- /etc/hosts

ローカルな名前解決

- /etc/nsswitch.conf

名前解決の順序などを設定。

(ローカルなhostsファイル→DNSサーバー)



```
[root@localhost ~]# ifconfig eth0
eth0   Link encap:Ethernet  HWaddr 00:0C:29:0A:7A:DD
       inet addr:192.168.140.129  Bcast:192.168.140.255  Mask:255.255.255
       (略)
```

eth0インターフェースに  
割り振られているIPアドレス  
などを確認

```
[root@localhost ~]# ping 192.168.140.2
PING 192.168.140.2 (192.168.140.2) 56(84) bytes of data.
64 bytes from 192.168.140.2: icmp_seq=1 ttl=128 time=0.209 ms
64 bytes from 192.168.140.2: icmp_seq=2 ttl=128 time=0.388 ms

--- 192.168.140.2 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 1191ms
rtt min/avg/max/mdev = 0.209/0.298/0.388/0.091 ms
```

ホスト192.168.140.2との疎  
通を確認する。  
Linuxのpingコマンドはデフォ  
ルトでパケットを送り続ける  
設定になっているため、  
ctrl+cで停止。

```
[root@localhost ~]# host www.lpi.or.jp
www.lpi.or.jp has address 202.218.212.222
```

www.lpi.or.jpを名前解決



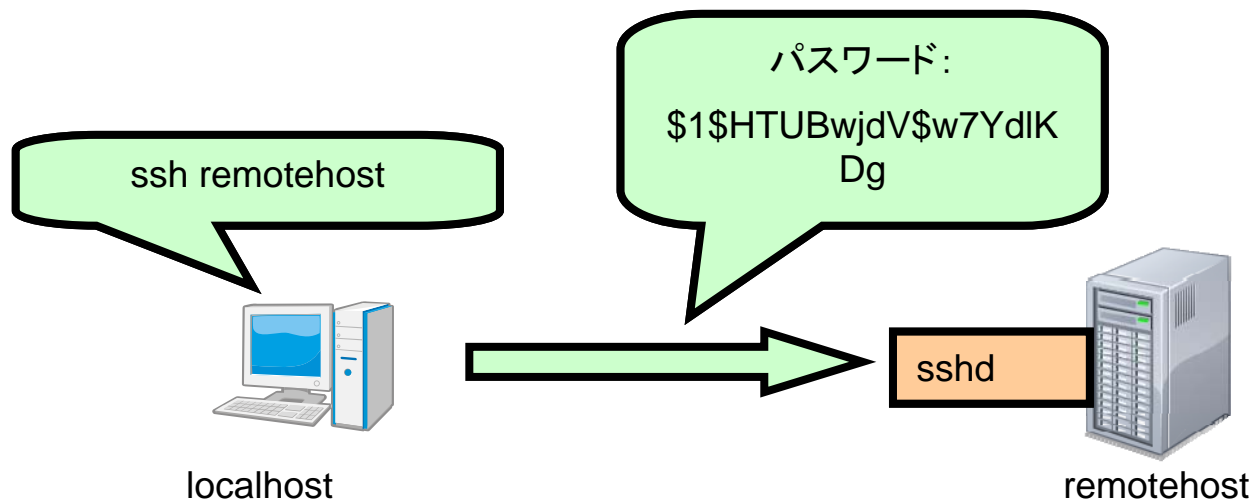
- 110.1 セキュリティ管理業務を実施する
- 110.2 ホストのセキュリティ設定
- 110.3 暗号化によるデータの保護





## ■ ssh (Secure Shell)

リモートホストに接続するためのプログラム。ネットワーク上を流れるデータは暗号化される。





- ホスト認証：接続先のホストのなりすましを防ぐ
- クライアント認証：信頼できる接続元であるかを確認
  - 公開鍵認証
  - パスワード認証

## ■ 暗号化方式

### ■ 共通鍵方式

暗号化も復号化も同じ鍵(共通鍵)

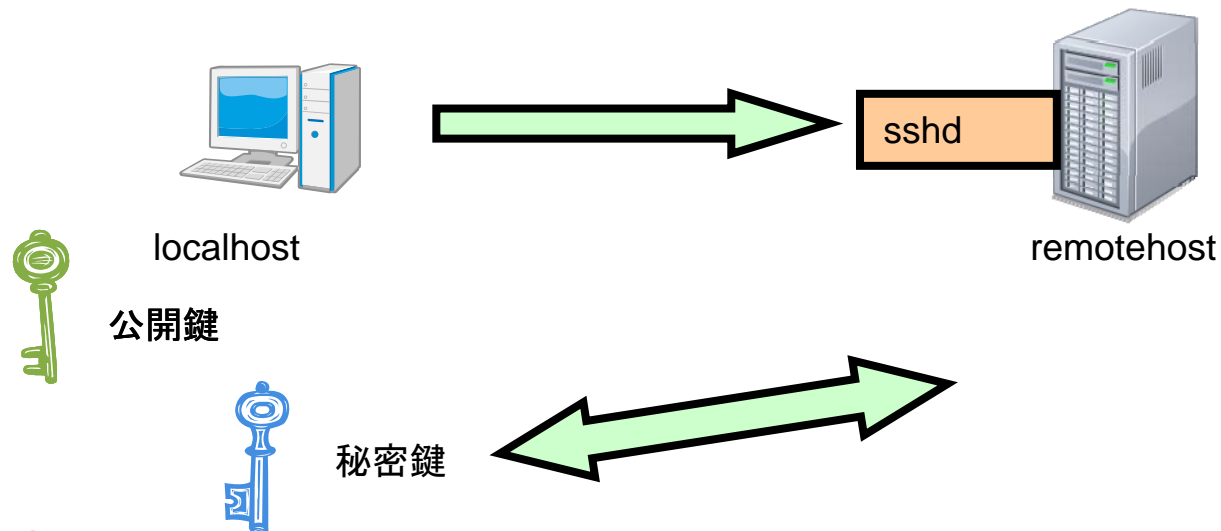
### ■ 公開鍵方式

暗号化と復号化を異なる鍵(公開鍵と秘密鍵)



```
[root@localhost ~]# ssh-keygen -t dsa
[root@localhost ~]# scp .ssh/id_dsa.pub remotehost:/root
[root@remotehost ~]# cat id_dsa.pub >> .ssh/authorized_keys
[root@localhost ~]# ssh remotehost
[root@remotehost ~]#
```

```
[root@remotehost ~]# vi /etc/ssh/sshd_config
PasswordAuthentication no
```





## Linux標準教科書

詳しくは下記URLで

<http://www.lpi.or.jp/linuxtext/text.shtml>

発行:エルピーアイジャパン



## Linux 教科書 LPICレベル1第4版

中島能和 (著) / 濱野賢一郎 (監修)

2009/05/12発売

出版社:翔泳社

552ページ

定価3,990円

ISBN- 9784798119311



## LPI認定試験 LPICレベル1《101/102》リリース3対応

橋本智裕、中島能和(監)

2009/3/26発行

出版社:秀和システム

608ページ

定価2,835円

ISBN- 978-4-7980-2229-1



## Linuxクックブック——Linuxを120%使いこなすレシピ集

Carla Schroder 著、林 秀幸 訳

2005/10発行

出版社: O'Reilly Japan

444ページ

定価3,780円

ISBN4-87311-248-6



質疑応答についてはお気軽にお声掛けください。  
ご清聴ありがとうございました。

**<http://www.skillbrain.co.jp>**



[info@skillbrain.co.jp](mailto:info@skillbrain.co.jp)