



ビデオ教材検索システム「Whale」
インストールマニュアル

目次

| | |
|---|----|
| 1. はじめに..... | 1 |
| 2. システム要件..... | 1 |
| 3. インストールに必要なもの..... | 1 |
| 4. インストール手順..... | 2 |
| 4.1. Whale 本体の設置..... | 2 |
| 4.2. 各種パッケージのインストール..... | 3 |
| 4.3. 設定..... | 6 |
| 4.3.1. Whale の設定..... | 6 |
| 4.3.2. Core の設定..... | 6 |
| 4.3.3. Web の設定..... | 6 |
| 4.4. その他のソフトウェアの設定..... | 7 |
| 4.4.1. PHP の設定..... | 7 |
| 4.4.2. MySQL の設定..... | 7 |
| 4.4.3. Perl の設定..... | 8 |
| 4.4.4. Cron の設定..... | 8 |
| 5. Q&A..... | 9 |
| 6. 付録..... | 10 |
| 6.1. Whale ツリー構造..... | 10 |
| 6.2. ipadic の utf-8 変換スクリプト (ipadic_conv_utf8.sh) | 12 |
| 6.3. データベース自動生成スクリプト (whale_database.sql) | 12 |

1. はじめに

本ドキュメントは、授業ビデオ検索システム「Whale」のインストールの手順を示したものです。

2. システム要件

以下にあげるシステム要件は、開発時の動作環境です。あくまでも例なので、その他の Linux ディストリビューションの環境でも知識があれば動作可能です。

| | |
|-----|------------------|
| OS | Ubuntu 11.10 |
| CPU | デュアルコア以上のものが望ましい |
| メモリ | 2GB 以上 |

3. インストールに必要なもの

- (1) Linux サーバ
- (2) Whale 本体 (Core+Web)
- (3) Apache (HTTP サーバ)
- (4) PHP
- (5) MySQL (データベース)
- (6) Julius (大語彙連続音声認識システム)
- (7) Darts (Chasen を動かすのに必要なライブラリ)
- (8) Chasen (形態素解析システム)
- (9) ipadic (Chasen 用の辞書)
- (10) FFmpeg (動画と音声の変換ソフト)

4. インストール手順

本章では Whale のインストールの手順について説明します。なお、本システムは Linux ディストリビューションのひとつである Ubuntu を用いて開発されたため、以下に示すコマンドは ubuntu 上で入力することを想定しています。(他のディストリビューションを用いる場合は、sudo の入力を省いたり、apt-get 等のコマンドを適切なものに変更したりしてインストール作業を進めることが可能)

今回は /whale ディレクトリ以下に Whale (ver130429) をインストールすると仮定して進めていくことにします。

4.1. Whale 本体の設置

本体を解凍し、公開したいディレクトリ上に設置します。また、ビデオのアップロード用のディレクトリを作成し、Core と Web にそのリンクを用意します。

まず whale_130429.tar.gz をブラウザまたは wget 等のコマンドを用いて、/home/(username)ディレクトリにダウンロードしてください。

以下のコマンドで解凍し、/whale に移動させます。これで配置は完了です。

```
$ tar zxvf whale_130429.tar.gz  
$ sudo mv whale_130429 /whale
```

4.2. 各種パッケージのインストール

Whale の動作に必要な各種パッケージをインストールします。すでにインストールされているパッケージがある場合はそのパッケージのインストールを飛ばしてください。

自動インストールスクリプト `install.sh` を用意してあるので、どうしてもインストールがわからない方は以下のコマンドを実行して各種パッケージをインストールください。スクリプトの内容は本節に記載してあるコマンドを順に実行するものとなっています。

```
$ cd /whale/util/  
$ sudo chmod +x install.sh  
$ sudo ./install.sh
```

(3) Apache (HTTP サーバ)

HTTP サーバである Apache をインストールします。

```
$ sudo apt-get install apache2
```

(4) PHP

PHP 本体とライブラリをインストールします。

```
$ sudo apt-get install php5 libapache2-mod-php5
```

(5) MySQL(データベース)

MySQL データベースとライブラリをインストールします。MySQL インストール時にパスワードを設定する画面が表示されますので、任意のパスワードに設定してください。

```
$ sudo apt-get install mysql-server  
$ sudo apt-get install php5-mysql
```

(6) Julius(大語彙連続音声認識システム)

Julius をインストールします。apt-get でもインストールできますが、本システムでは使用することができませんので、以下の手順に従ってインストールしてください。

ブラウザ等から `julius-4.2.2.tar.gz` のアーカイブファイルをダウンロードして home ディレクトリにおいてください。

*どうしてもコマンドラインしか使用できない方は `wget` を利用してください。ただし、バージョンにより URL が変わるのでご注意ください。

```
$ wget http://ij.dl.sourceforge.jp/julius/56549/julius-4.2.2.tar.gz
```

ホームディレクトリに配置した場合は、まずは以下のコマンドで展開します。

```
$ tar xzf julius-4.2.2.tar.gz  
$ cd julius-4.2.2
```

本システムで使用するために Julius のパラメータを変更してからコンパイル及びインストールを行います。sed コマンドで置換を行っていますが、うまくいかない場合はテキストエディタで直接変

更してください。

```
$sed -e "s/#define MAXSEQNUM          150/#define MAXSEQNUM          10000/"  
libsent/include/sent/speech.h > speech.new  
$ mv speech.new libsent/include/sent/speech.h  
$ sudo ./configure  
$ sudo make  
$ sudo make install
```

(7) Darts (Chasen を動かすのに必要なライブラリ)

Darts をインストールします。Chasen より必ず先にインストールしてください。ブラウザ等から darts-0.32.tar.gz のアーカイブファイルをダウンロードして home ディレクトリにおいてください。Darts のインストールの途中でコンパイルするためのコンパイラをインストールします。

*どうしてもコマンドラインしか使用できない方は wget を利用してください。ただし、バージョンにより URL が変わるのでご注意ください。

```
$ wget http://chasen.org/~taku/software/darts/src/darts-0.32.tar.gz
```

ホームディレクトリに配置した場合は、以下のコマンドによりインストールします。

```
$ tar xzf darts-0.32.tar.gz  
$ cd darts-0.32  
$ sudo apt-get install build-essential  
$ sudo ./configure  
$ sudo make  
$ sudo make check  
$ sudo make install
```

(8) Chasen (形態素解析システム)

Chasen をインストールします。ブラウザ等から chasen-2.4.5.tar.gz のアーカイブファイルをダウンロードして home ディレクトリにおいてください。

*どうしてもコマンドラインしか使用できない方は wget を利用してください。ただし、バージョンにより URL が変わるのでご注意ください。

```
$ wget http://ij.dl.sourceforge.jp/chasen-legacy/56305/chasen-2.4.5.tar.gz
```

ホームディレクトリに配置した場合は、以下のコマンドによりインストールします。

```
$ tar xzf chasen-2.4.5.tar.gz  
$ cd chasen-2.4.5  
$ sudo ./configure  
$ sudo make  
$ sudo make install
```

(9) ipadic (Chasen 用の辞書) ,

ipadic をインストールします。ブラウザ等から ipadic-2.7.0.tar.gz の archive ファイルをダウンロードして home ディレクトリにおいてください。

*どうしてもコマンドラインしか使用できない方は wget を利用してください。ただし、バージョンにより URL が変わるので、ご注意ください。

```
$ wget http://jaist.dl.sourceforge.jp/ipadic/24435/ipadic-2.7.0.tar.gz
```

本辞書は文字コードが Whale のシステムと対応していないため文字コードを変換したのちにインストールを行います。なお変換スクリプトは whale/util にありますのでご利用ください。変換スクリプトファイルの作成方法は、付録をご参照ください。

```
$ tar xzf ipadic-2.7.0.tar.gz
$ cd ipadic-2.7.0
$ sudo ./configure
$ cp /whale/util/ipadic_conv_utf8.sh ./
$ chmod +x ipadic_conv_utf8.sh
$ ./ipadic_conv_utf8.sh
$ sudo ldconfig
$ `chasen-config --mkchadic`/makemat -i w
$ `chasen-config --mkchadic`/makeda -i w chadic *.dic
$ sudo make install
$ cd /usr/local/etc/
$ sudo sh -c "iconv -f euc-jp -t utf-8 chasenrc > temp"
$ sudo mv temp chasenrc
```

(3) FFmpeg (動画と音声の変換ソフト)

FFmpeg をインストールします。

```
$ sudo apt-get install ffmpeg
```

4.3. 設定

Whale の設定を行います。Whale 全体の設定ファイルの変更と core および web ディレクトリにパーミッションやシンボリックリンク等の設定を行います。なお、ここからは whale の配置を/whale としていることを前提としていますので、そのほかの場所に配置した場合は、コマンドのファイルパス等を適切に変えて進めてください。

4.3.1. Whale の設定

Whale の設定を行います。

whale.conf の設定（場所は/whale/core/whale.conf）

| 設定名 | 設定例 | 説明 |
|-------------|--------------|-----------------|
| coredir | /whale/core | core ディレクトリのパス |
| webdir | /whale/web | web ディレクトリのパス |
| videodir | /whale/video | video ディレクトリのパス |
| workdirname | workspace | 音声認識結果保存ディレクトリ名 |
| db | whale | システムのデータベース名 |
| dbuser | whale | データベースのユーザ名 |
| dbpass | whaleadmin | データベースのパスワード |

4.3.2. Core の設定

Whale の core ディレクトリの各ファイルやディレクトリにパーミッションを設定します。Core と Web の設定は自動設定スクリプト config.sh を用意してありますのでご利用ください。

```
$ cd /whale/util/  
$ sudo chmod +x config.sh  
$ sudo ./config.sh
```

```
$ cd /whale/core  
$ sudo chmod +x *.pl  
$ sudo chmod +x wavcut  
$ sudo chmod +x wav_timedata  
$ sudo chmod +x td  
$ sudo chmod 777 workspace
```

4.3.3. Web の設定

Whale の web ディレクトリ内の各ファイルやディレクトリにパーミッションを設定します。また、core ディレクトリの workspace ディレクトリのシンボリックリンク等を web に作成します。4.3.2 で自動設定スクリプトを利用した場合は、以下の設定もされていますので飛ばしてください。

```
$ cd /whale/web  
$ sudo chmod +x *.pl
```



```
$ sudo chmod +x hb4/hb4
$ sudo chmod +x hb4_tfidf_norm/hb4_tfidf_norm
$ sudo chmod +x tf_idf_hb4_search/tf_idf_hb4_search
$ sudo chmod 777 temporary
$ sudo chmod 777 video
$ ln -s /whale/video/ video
$ ln -s /whale/core/workspace/ workspace
```

4.4. その他のソフトウェアの設定

その他のソフトウェアに関して設定を行います。バージョンによって、設定項目がなくなったり変更されたりしている場合があります。これらの設定は、自動設定スクリプトには含まれてはいませんので手動で設定を行ってください。

4.4.1. PHP の設定

php.ini の設定（場所は/etc/php5/apache2/php.ini）

該当する以下の部分を修正してください。ファイルのバックアップを取ってから設定することをお勧めします。また、以下の設定は最低限の値となっていますので、使用する環境に合わせて数値を各自設定してください

| 修正前 | 修正後 |
|--------------------------|-----------------------------|
| max_execution_time = 30 | max_execution_time = 300 |
| memory_limit = 128MB | memory_limit = 2048MB |
| post_max_size = 8MB | post_max_size = 2048MB |
| upload_max_filesize = 2M | upload_max_filesize = 2048M |

4.4.2. MySQL の設定

```
$ mysql -u root -p
```

ここから MySQL のコマンド操作となります。以下のコマンドを記述して新しいユーザ（今回は whale とする）を追加してください。またパスワード（今回は whaleadmin とする）も設定してください。

```
mysql> grant all ON *.* to whale@"localhost" identified by "whaleadmin";
mysql> flush privileges;
mysql> quit
```

追加したユーザでログインしなおします。

```
$ mysql -u whale -p
```

ログインできればユーザ登録が成功しています。次に MySQL のコマンドから whale 専用のデータベースの作成とテーブルを追加します。スクリプトにすべて記述してあるので、スクリプトを読み込んでください。また、初期の whale 管理ユーザが自動的に追加されます。（user:whale, pass:whaleadmin）

```
mysql> source /whale/util/whale_database.sql  
mysql> quit
```

4.4.3. Perl の設定

システムの動作に必要なモジュールを追加します。

```
$ sudo perl -MCPAN -e shell
```

初めて起動した場合は設定画面が出ますが、最初の項目で自動設定にするか聞かれるので、適当に Enter を打っていれば設定が終わるはずです。設定後 CPAN のコマンドラインになるので、モジュールをインストールします。CPAN のコマンドラインにならなかった場合は、上記のコマンドをもう一度入力して CPAN コマンドラインを起動させてください。

```
cpan> install Config::Simple  
cpan> exit
```

4.4.4. Cron の設定

Cron と呼ばれるスクリプトを自動実行させるソフトウェアに Whale を設定します。以下のコマンドを入力して設定ファイルを開いてください。

```
$ crontab -e
```

何らかのエディタで開かれたはずですが、この設定ファイルに以下のコマンドを追記してください。

```
00 4 * * * perl /whale/core/cron_whale.pl > /dev/null 2>&1
```

上記の設定では、毎朝 4 時に Whale が自動的に未解析のビデオファイルを解析するように設定されます。時間を変えたい場合は Cron の設定方法に従って設定を書き換えてください。

5. アンインストール

Whale のアンインストールを行う場合は、以下の手順に沿って操作してください。

1. /whale ディレクトリを削除
2. MySQL の whale データベースおよび whale ユーザを削除
3. Whale システム用に cron に登録した記述を削除
4. Whale インストール時にインストールした各種パッケージをアンインストール（不要な場合）

6. Q&A

| |
|--|
| Q1 : Whale のインストールに失敗する。 |
| A1 : 次の 2 点の問題が考えられます。①インストールの手順がおかしい可能性があります。②Whale 以外にインストールしたソフトウェアのバージョンが適切ではない可能性があります。 |
| Q2 : Ubuntu 以外のディストリビューションで使いたい。 |
| A2 : Ubuntu 独自で動作するものは使ってはいないはずなので、おそらくほかのディストリビューションで運用することも可能です。しかし、本ドキュメントでは Ubuntu にインストールする方法しか説明してありませんので、そのディストリビューションにあったインストール方法にコマンドを置き換えて、インストールを行ってください。 |
| Q3 : Whale って何？くじらですか？ |
| A3 : くじらです。本システムの作成者がなんとなく思いついた名前です。 |

7. 付録

細かいスクリプトの内容はここに記載しておきますので、何らかの形でうまくいかない方は参照してください。自分で作成する場合は適当なエディタを起動させて、以下のコードを記述して保存してください。

7.1. Whale ツリー構造

インストールに成功したらおそらくこのようなツリー構造になります。何か動作がおかしい場合などに御参考ください。

```
/whale
├── [drwxrwxr-x] core
│   ├── [-rwxrwxr-x] all.pl
│   ├── [-rwxrwxr-x] chasen_jiritsu.pl
│   ├── [-rw-rw-r--] chasen_stopword.txt
│   ├── [-rw-rw-r--] check.txt
│   ├── [-rwxrwxr-x] cron_whale.pl
│   ├── [drwxrwxr-x] cut
│   ├── [-rwxrwxr-x] cut_scene.pl
│   ├── [-rwxrwxr-x] cut_triphone.pl
│   ├── [-rwxrwxr-x] database.pl
│   ├── [-rwxrwxr-x] endian.pl
│   ├── [-rwxrwxr-x] get_sentence.pl
│   ├── [-rwxrwxr-x] get_triphone.pl
│   ├── [-rwxrwxr-x] julius.pl
│   ├── [drwxrwxr-x] julius_config
│   ├── [drwxrwxr-x] CSJ-APS
│   │   ├── [-rw-rw-r--] hmmdefs
│   │   └── [-rw-rw-r--] hmmdefs.gz
│   ├── [-rw-rw-r--] kyoto-gt1_19k.dic
│   ├── [-rw-rw-r--] kyoto_kouen.bigram
│   └── [-rw-rw-r--] logicalTri.add
│   ├── [-rwxrwxrwx] td
│   ├── [drwxrwxr-x] vad
│   │   ├── [-rw-rw-r--] readme.txt
│   │   ├── [-rw-rw-r--] time.dat
│   │   ├── [-rw-rw-r--] vad
│   │   └── [-rw-rw-r--] vad.c
│   ├── [-rwxrwxr-x] wav_timedata
│   ├── [-rwxrwxr-x] wavcut
│   └── [drwxrwxrwx] workspace
├── [-rw-rw-r--] tree.txt
├── [drwxrwxr-x] util
│   ├── [-rw-rw-r--] config.sh
│   ├── [-rwxrwxr-x] install.sh
│   ├── [-rw-rw-r--] ipadic_conv_utf8.sh
│   └── [-rw-rw-r--] whale_database.sql
├── [drwxrwxrwx] video
├── [drwxrwxr-x] web
│   ├── [-rw-rw-r--] admin.php
│   ├── [-rw-rw-r--] api_favorite.php
│   ├── [-rw-rw-r--] api_memo.php
│   ├── [-rw-rw-r--] api_util.php
│   ├── [drwxrwxr-x] css
│   │   ├── [-rw-rw-r--] admin.css
│   └── [drwxrwxr-x] images
```

| | | |
|----|--------------|-----------------------------|
| -- | [-rw-rw-r--] | ajax-loader.gif |
| -- | [-rw-rw-r--] | ajax-loader.png |
| -- | [-rw-rw-r--] | icons-18-black.png |
| -- | [-rw-rw-r--] | icons-18-white.png |
| -- | [-rw-rw-r--] | icons-36-black.png |
| -- | [-rw-rw-r--] | icons-36-white.png |
| -- | [-rw-rw-r--] | index.css |
| -- | [-rw-rw-r--] | jquery.mobile-1.1.1.min.css |
| -- | [-rw-rw-r--] | main.css |
| -- | [-rw-rw-r--] | register.css |
| -- | [-rw-rw-r--] | resultview.css |
| -- | [-rw-rw-r--] | upload.css |
| -- | [-rw-rw-r--] | video.css |
| -- | [-rw-rw-r--] | datelist.php |
| -- | [drwxrwxr-x] | hb4 |
| -- | [-rw-rw-r--] | 00readme |
| -- | [-rw-rw-r--] | compile |
| -- | [-rw-rw-r--] | hash.c |
| -- | [-rw-rw-r--] | hash.h |
| -- | [-rwxrwxr-x] | hb4 |
| -- | [-rw-rw-r--] | hb4.cpp |
| -- | [drwxrwxr-x] | r |
| -- | [-rw-rw-r--] | hb4.out_col |
| -- | [-rw-rw-r--] | hb4.out_row |
| -- | [-rw-rw-r--] | hb4.outf |
| -- | [-rw-rw-r--] | hb4.term |
| -- | [drwxrwxr-x] | hb4_tfidf_norm |
| -- | [-rw-rw-r--] | 00readme |
| -- | [-rw-rw-r--] | compile |
| -- | [-rwxrwxr-x] | hb4_tfidf_norm |
| -- | [-rw-rw-r--] | hb4_tfidf_norm.cpp |
| -- | [-rw-rw-r--] | in3.txt |
| -- | [drwxrwxr-x] | r |
| -- | [-rw-rw-r--] | hb4.out_col |
| -- | [-rw-rw-r--] | hb4.out_row |
| -- | [-rw-rw-r--] | hb4.outf |
| -- | [-rw-rw-r--] | hb4.term |
| -- | [-rw-rw-r--] | hb4.tfidf |
| -- | [-rw-rw-r--] | hb4.tfidf_norm |
| -- | [-rw-rw-r--] | tfidf.mat |
| -- | [-rw-rw-r--] | tfidf.txt |
| -- | [-rw-rw-r--] | tfidf_norm.xlsx |
| -- | [-rw-rw-r--] | index.php |
| -- | [drwxrwxr-x] | js |
| -- | [-rw-rw-r--] | index.js |
| -- | [-rw-rw-r--] | jqm_custom.js |
| -- | [-rw-rw-r--] | jquery-1.7.1.min.js |
| -- | [-rw-rw-r--] | jquery.cookie.js |
| -- | [-rw-rw-r--] | jquery.mobile-1.1.1.js |
| -- | [-rw-rw-r--] | jquery.mobile-1.1.1.min.js |
| -- | [-rw-rw-r--] | video.js |
| -- | [-rw-rw-r--] | kujira.ico |
| -- | [-rw-rw-r--] | kujira_apple.png |
| -- | [-rw-rw-r--] | login.php |
| -- | [-rw-rw-r--] | mysql_manager.php |
| -- | [-rw-rw-r--] | register.php |
| -- | [-rw-rw-r--] | registered.php |
| -- | [-rwxrwxr-x] | scene_table.pl |

```
| |--- [-rw-rw-r--] search.php
| |--- [-rwxrwxr-x] search.pl
| |--- [drwxrwxr-x] temporary
| |--- [drwxrwxr-x] tf_idf_hb4_search
| |   |--- [-rw-rw-r--] Readme.bak
| |   |--- [-rw-rw-r--] cdp.c
| |   |--- [-rw-rw-r--] cdp.h
| |   |--- [-rw-rw-r--] get_token.c
| |   |--- [-rw-rw-r--] get_token.h
| |   |--- [-rw-rw-r--] kana.txt
| |   |--- [-rw-rw-r--] key2n.txt
| |   |--- [-rw-rw-r--] key2tri
| |   |--- [-rw-rw-r--] main.c
| |   |--- [-rw-rw-r--] net.txt
| |   |--- [-rw-rw-r--] readme.txt
| |   |--- [-rw-rw-r--] rom.txt
| |   |--- [drwxrwxr-x] sh
| |     |--- [-rw-rw-r--] kana2rom.pl
| |     |--- [-rw-rw-r--] rom2tri
| |     |--- [-rw-rw-r--] rom2tri.bak
| |     |--- [-rw-rw-r--] rom2tri.de
| |   |--- [-rw-rw-r--] sub_hash.c
| |   |--- [-rw-rw-r--] sub_hash.h
| |   |--- [-rw-rw-r--] tf_idf.c
| |   |--- [-rw-rw-r--] tf_idf.h
| |   |--- [-rw-rw-r--] tf_idf_hb4_kensaku_new_mod
| |   |--- [-rw-rw-r--] tf_idf_hb4_search
| |   |--- [-rw-rw-r--] topic.tri
| |   |--- [-rw-rw-r--] tri.txt
| |--- [lrwxrwxrwx] video -> /whale/video/
| |--- [-rw-rw-r--] video.php
| |--- [lrwxrwxrwx] workspace -> /whale/core/workspace/
|--- [-rw-rw-r--] whale.conf
```

7.2. ipadic の utf-8 変換スクリプト (ipadic_conv_utf8.sh)

```
#!/bin/sh
for file in *.dic *.cha
do
if [ -f $file ]; then
echo $file
iconv -f euc-jp -t utf-8 $file > tmpfile
mv tmpfile $file
fi
done
exit
```

7.3. データベース自動生成スクリプト (whale_database.sql)

```
create database whale default character set utf8;
use whale

create table class(
id integer not null primary key auto_increment,
grade integer,
subject_id integer,
teacher_id integer,
```

```
year integer not null
);

create table favorite(
id integer not null primary key auto_increment,
user_name text not null,
video_id integer not null
);

create table memo(
id integer not null primary key auto_increment,
user_name text not null,
video_id integer not null,
memo text not null
);

create table subject(
id integer not null primary key auto_increment,
name text,
identifier text
);

create table teacher(
id integer not null primary key auto_increment,
name text,
identifier text
);

create table user(
id integer not null primary key auto_increment,
name text not null,
pass text,
admin int not null
);

create table video(
id integer not null primary key auto_increment,
name text ,
class_id integer,
date text,
recognition integer not null
);
```