

# FreeBSDでOpen JTalkを使う

音声合成ソフトウェアの Open JTalk を FreeBSD で実行させるまでの話。

2017/03/17

[Ports作りました](#)

2014年10月06日

昔やりかけてた Open JTalk のコンパイルを再挑戦。今回は音声合成までいけた。

今回の環境は、FreeBSD/AMD64 10.0 RELEASEで、インテルのCPUが搭載されたマシンにAMDのはついに寿命がきてしまいました。

2015年6月21日

open\_jtalk-1.08だと以下でコンパイルが通るそう。

[https://twitter.com/s\\_mitu/status/589726270952382465](https://twitter.com/s_mitu/status/589726270952382465)

@namaedayo #FreeBSDworkshop open\_jtalk-1.08.mod/configure のICONV\_CONST /\*\*/を  
ICONV\_CONST const に書き換えるでビルドできたと思います。

— 鈴木光宏 (@s\_mitu) 2015, 4月 19

## 参照情報

- [Open JTalk - HMM-based Text-to-Speech System](#)
- [mmdagent.jp](#)
- [Open JTalk - PIB](#)
- [Open JTalk で音声合成して日本語テキスト読み上げ - ぼち \\* ぷ?ち](#)
- [takayanの雑記帳](#)
- [サーバに日本語を解析させてしゃべらせる:sousu no memo](#)
- [ONScripter のページ:iconvの引数のコンパイルエラー](#)

## 必要なファイル群の収集

Open JTalkは

- Open JTalk - 本体
- hts\_engine - 音声合成エンジン
- HTS voice - 合成する音声情報
- Dictionary - 日本語解析のための辞書

で構成されます。

## 本体、辞書、サンプル音声情報

<http://sourceforge.net/projects/open-jtalk/files/> より、

- [open\\_jtalk-1.07.tar.gz](#)

- `hts_voice_nitech_jp_atr503_m001-1.05.tar.gz`
- `open_jtalk_dic_utf_8-1.07.tar.gz`

を取得し、作業用ディレクトリへコピー、展開しておきます。

## 音声合成エンジン

<http://sourceforge.net/projects/hts-engine/files/> より、

- `hts_engine_API-1.08.tar.gz`

を取得し、作業用ディレクトリへコピー、展開してきます。

## サンプル音声（話者Mei）

<http://sourceforge.net/projects/mmdagent/files/> より、

- `MMDAgent_Example-1.4.zip`

を取得し、作業用ディレクトリへコピー、展開してきます。MMDAgent用の話者に女性（話者:Mei）の音声情報が同梱されています。

## コンパイルの前提

- FreeBSDでデフォルトのclangではコンパイルができないのでgccを導入しておく必要があります。筆者はgcc4.8をportsから導入しました。gmakeも使える必要があります。
- libiconvを使っているため、こちらを導入する必要があります。
- Open JTalkのビルドの仕方によっては、既存のMecabの辞書を上書きするかもしれません。再インストールの準備が必要です。

## コンパイル

### hts\_engine\_API

展開したディレクトリ `hts_engine_API-1.08` で以下のコマンドラインを実行します。

```
root@amane:/home/k896951/openjtalk/hts_engine_API-1.08 # ./configure --with-charset=UTF-8 LDFLAGS="-Wl,-rpath=/usr/local/lib/gcc48" CC=gcc48 CXX=c++48
```

hts\_engineのコンパイルに際しては `LDFLAGS="-Wl,-rpath=/usr/local/lib/gcc48"` の記述が無くても大丈夫なようです。

これは、リンケージの際にgcc4.8のものを使うよ！という宣言ですね。

次に、HTS\_misc.c に修正を加えます。FreeBSDでも `fpos_t` は構造体じゃないようです。あとAndroidでもコンパイルできるのかな？動きそうな感じがしますよねこのソースだと。

```
root@amane:/home/amane/openjtalk # diff -c
./hts_engine_API-1.08/lib/HTS_misc.c.org
./hts_engine_API-1.08/lib/HTS_misc.c
*** ./hts_engine_API-1.08/lib/HTS_misc.c.org      2014-10-06
03:55:37.000000000 +0900
--- ./hts_engine_API-1.08/lib/HTS_misc.c          2014-10-06
03:56:18.000000000 +0900
*****
*** 245,251 ****
    } else if (fp->type == HTS_FILE) {
        fpos_t pos;
        fgetpos((FILE *) fp->pointer, &pos);
! #if defined(_WIN32) || defined(__CYGWIN__) || defined(__APPLE__) ||
defined(__ANDROID__)
        return (size_t) pos;
    #else
        return (size_t) pos.__pos;
--- 245,251 ----
    } else if (fp->type == HTS_FILE) {
        fpos_t pos;
        fgetpos((FILE *) fp->pointer, &pos);
! #if defined(_WIN32) || defined(__CYGWIN__) || defined(__APPLE__) ||
defined(__ANDROID__) || defined(__FreeBSD__)
        return (size_t) pos;
    #else
        return (size_t) pos.__pos;
root@amane:/home/amane/openjtalk #
```

あとはコンパイルとインストール。こちらは結構すんなりといけます。

```
root@amane:/home/amane/openjtalk/hts_engine_API-1.08 # gmake
<中略>
root@amane:/home/amane/openjtalk/hts_engine_API-1.08 # gmake install
Making install in lib
gmake[1]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/lib' に入ります
gmake[2]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/lib' に入ります
test -z "/usr/local/lib" || ../config/install-sh -c -d "/usr/local/lib"
/usr/bin/install -c -m 644 'libHTSEngine.a' '/usr/local/lib/libHTSEngine.a'
ranlib '/usr/local/lib/libHTSEngine.a'
gmake[2]: `install-data-am' に対して行うべき事はありません。
gmake[2]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/lib' から出ます
gmake[1]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/lib' から出ます
Making install in bin
gmake[1]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/bin' に入ります
gmake[2]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/bin' に入ります
test -z "/usr/local/bin" || ../config/install-sh -c -d "/usr/local/bin"
```

```
/usr/bin/install -c 'hts_engine' '/usr/local/bin/hts_engine'  
gmake[2]: `install-data-am' に対して行うべき事はありません。  
gmake[2]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/bin' から出ま  
す  
gmake[1]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08/bin' から出ま  
す  
gmake[1]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08' に入ります  
gmake[2]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08' に入ります  
gmake[2]: `install-exec-am' に対して行うべき事はありません。  
test -z "/usr/local/include" || config/install-sh -c -d "/usr/local/include"  
/usr/bin/install -c -m 644 'include/HTS_engine.h'  
'/usr/local/include/HTS_engine.h'  
gmake[2]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08' から出ます  
gmake[1]: ディレクトリ `/home/amane/openjtalk/hts_engine_API-1.08' から出ます  
root@amane:/home/amane/openjtalk/hts_engine_API-1.08 #
```

## open\_jtalk

展開したディレクトリ open\_jtalk-1.07 で以下のコマンドラインを実行します。

```
root@amane:/home/amane/openjtalk/open_jtalk-1.07 # ./configure --with-  
charset=UTF-8 LDFLAGS="-Wl,-rpath=/usr/local/lib/gcc48" CC=gcc48 CXX=c++48
```

open\_jtalkのコンパイルに際しては LDFLAGS="-Wl,-rpath=/usr/local/lib/gcc48" の記述は必須です。こ  
れが無いと、コンパイル時に

```
Making all in mecab-naist-jdic  
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07/mecab-naist-  
jdic' に入ります  
../mecab/src/mecab-dict-index -d . -o . -f EUC-JP -t utf-8  
/usr/local/lib/compat/libstdc++.so.6: version GLIBCXX_3.4.11 required by  
/home/amane/openjtalk/open_jtalk-1.07/mecab/src/mecab-dict-index not found  
gmake[1]: *** [char.bin] エラー 1  
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07/mecab-naist-  
jdic' から出ます  
gmake: *** [all-recursive] エラー 1  
root@amane:/home/amane/openjtalk/open_jtalk-1.07 #
```

のようなエラーが最後に発生してしまいます。

次に、iconv\_utils.cpp に修正を加えます。ICONV\_CONST の値が定義されていないので iconv() の第二引  
数の型が const char \*\* にならないようです。結果 libiconv でインストールしたヘッダの定義とあわず  
コンパイルエラーとなります。  
おそらく Mecab に喰わせる文字列変換の処理なのでしょう。面倒なので ICONV\_CONST を const へ書  
き換えました。

```
root@amane:/home/amane/openjtalk # diff -c  
open_jtalk-1.07/mecab/src/iconv_utils.cpp.org  
open_jtalk-1.07/mecab/src/iconv_utils.cpp  
*** open_jtalk-1.07/mecab/src/iconv_utils.cpp.org      2014-10-06
```

```
05:32:34.000000000 +0900
--- open_jtalk-1.07/mecab/src/iconv_utils.cpp 2014-10-06
05:33:06.000000000 +0900
*****
*** 179,185 ***
    size_t olen_org = olen;
    iconv(ic_, 0, &ilen, 0, &olen); // reset iconv state
    while (ilen != 0) {
!       if (iconv(ic_, (ICONV_CONST char *)&ibuf, &ilen, &obuf, &olen)
            == (size_t) -1) {
                return false;
            }
--- 179,185 ----
    size_t olen_org = olen;
    iconv(ic_, 0, &ilen, 0, &olen); // reset iconv state
    while (ilen != 0) {
!       if (iconv(ic_, (const char *)&ibuf, &ilen, &obuf, &olen)
            == (size_t) -1) {
                return false;
            }
root@amane:/home/amane/openjtalk #
```

あとはコンパイルとインストール。

```
root@amane:/home/amane/openjtalk/hts_engine_API-1.08 # gmake
<中略>
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07/mecab-naist-jdic' に入ります
../mecab/src/mecab-dict-index -d . -o . -f EUC-JP -t utf-8
reading ./unk.def ... 40
emitting double-array: 100% |#####|
./model.def is not found. skipped.
reading ./naist-jdic.csv ... 481837
emitting double-array: 100% |#####|
reading ./matrix.def ... 1377x1377
emitting matrix      : 100% |#####|

done!
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07/mecab-naist-jdic' から出ます
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07' に入ります
gmake[1]: `all-am' に対して行うべき事はありません。
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07' から出ます
root@amane:/home/amane/openjtalk/open_jtalk-1.07 #
root@amane:/home/amane/openjtalk/open_jtalk-1.07 # gmake install
<中略>
gmake[2]: `install-exec-am' に対して行うべき事はありません。
test -z "/usr/local/dic" || ../config/install-sh -c -d "/usr/local/dic"
/usr/bin/install -c -m 644 'char.bin' '/usr/local/dic/char.bin'
/usr/bin/install -c -m 644 'matrix.bin' '/usr/local/dic/matrix.bin'
/usr/bin/install -c -m 644 'sys.dic' '/usr/local/dic/sys.dic'
```

```
/usr/bin/install -c -m 644 'unk.dic' '/usr/local/dic/unk.dic'  
/usr/bin/install -c -m 644 'left-id.def' '/usr/local/dic/left-id.def'  
/usr/bin/install -c -m 644 'right-id.def' '/usr/local/dic/right-id.def'  
/usr/bin/install -c -m 644 'rewrite.def' '/usr/local/dic/rewrite.def'  
/usr/bin/install -c -m 644 'pos-id.def' '/usr/local/dic/pos-id.def'  
gmake[2]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07/mecab-naist-  
jdic' から出ます  
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07/mecab-naist-  
jdic' から出ます  
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07' に入ります  
gmake[2]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07' に入ります  
gmake[2]: `install-exec-am' に対して行うべき事はありません。  
gmake[2]: `install-data-am' に対して行うべき事はありません。  
gmake[2]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07' から出ます  
gmake[1]: ディレクトリ `/home/amane/openjtalk/open_jtalk-1.07' から出ます  
root@amane:/home/amane/openjtalk/open_jtalk-1.07 #
```

もし `/usr/local/dic/` にMecabの辞書をインストールしていたのであれば、これによってファイルが上書きさますのでMecabの再インストールをするか `configure` の実行時にインストールパスを明示指定して上書きされないように調整してください。筆者環境ではMecabは別ディレクトリにインストールされているので問題はありませんでした。

## Open JTalkを実行してみる

```
root@amane:/home/amane/openjtalk # open_jtalk  
The Japanese TTS System "Open JTalk"  
Version 1.07 (http://open-jtalk.sourceforge.net/)  
Copyright (C) 2008-2013 Nagoya Institute of Technology  
All rights reserved.  
  
The HMM-Based Speech Synthesis Engine "hts_engine API"  
Version 1.08 (http://hts-engine.sourceforge.net/)  
Copyright (C) 2001-2013 Nagoya Institute of Technology  
2001-2008 Tokyo Institute of Technology  
All rights reserved.  
  
Yet Another Part-of-Speech and Morphological Analyzer "Mecab"  
Version 0.994 (http://mecab.sourceforge.net/)  
Copyright (C) 2001-2008 Taku Kudo  
2004-2008 Nippon Telegraph and Telephone Corporation  
All rights reserved.  
  
NAIST Japanese Dictionary  
Version 0.6.1-20090630 (http://naist-jdic.sourceforge.jp/)  
Copyright (C) 2009 Nara Institute of Science and Technology  
All rights reserved.  
  
open_jtalk - The Japanese TTS system "Open JTalk"
```

```

usage:
  open_jtalk [ options ] [ infile ]
options:
[ def][ min-- max]
  -x dir      : dictionary directory
[ N/A]
  -m htsvoice : HTS voice files
[ N/A]
  -ow s       : filename of output wav audio (generated speech)
[ N/A]
  -ot s       : filename of output trace information
[ N/A]
  -s i        : sampling frequency
[ auto][ 1-- ]
  -p i        : frame period (point)
[ auto][ 1-- ]
  -a f        : all-pass constant
[ auto][ 0.0-- 1.0]
  -b f        : postfiltering coefficient
[ 0.0][ 0.0-- 1.0]
  -r f        : speech speed rate
[ 1.0][ 0.0-- ]
  -fm f       : additional half-tone
[ 0.0][ -- ]
  -u f        : voiced/unvoiced threshold
[ 0.5][ 0.0-- 1.0]
  -jm f       : weight of GV for spectrum
[ 1.0][ 0.0-- ]
  -jf f       : weight of GV for log F0
[ 1.0][ 0.0-- ]
  -z i        : audio buffer size (if i==0, turn off)
[ 0][ 0-- ]
  infile:
    text file
[stdin]

root@amane:/home/amane/openjtalk #

```

実は難しいオプションはいりません。hts\_voice\_nitech\_jp\_atr503\_m001-1.05ディレクトリのINSTALLを参照するとコマンドラインの説明がありました。

```

root@amane:/home/amane/openjtalk/hts_voice_nitech_jp_atr503_m001-1.05 # cat
INSTALL
Installation Instructions
*****

1. After unpacking the tar.gz file, cd to the HTS voice directory.

2. Run engine with appropriate options.

% hts_engine -m nitech_jp_atr503_m001.htsvoice \

```

```
-ow output.wav input.lab

% open_jtalk -m nitech_jp_atr503_m001.htsvoice \
  -ow output.wav -x dic_dir input.txt
root@amane:/home/amane/openjtalk/hts_voice_nitech_jp_atr503_m001-1.05 #
```

-m オプションで音声情報 `-x` オプションで辞書を指定するだけでとりあえず音声合成できます。

```
root@amane:/home/amane/openjtalk # cat test.txt|nkf
今日は台風がきていますね。電車の運行状況が、少し気になります。

root@amane:/home/amane/openjtalk # open_jtalk -m
MMDAgent_Example-1.4/Voice/mei/mei_normal.htsvoice -ow outme.wav -x
open_jtalk_dic_utf_8-1.07 test.txt
root@amane:/home/amane/openjtalk # open_jtalk -m
hts_voice_nitech_jp_atr503_m001-1.05/nitech_jp_atr503_m001.htsvoice -ow
outfe.wav -x open_jtalk_dic_utf_8-1.07 test.txt
root@amane:/home/amane/openjtalk # ll *wav
-rw-r--r--  1 root  custom  593804 10月  6 03:07 outfe.wav
-rw-r--r--  1 root  custom  565004 10月  6 03:07 outme.wav
root@amane:/home/amane/openjtalk #
```

test.txt はUTF-8で記述したテキストファイルです `open_jtalk_dic_utf_8-1.07`ディレクトリにはUTF-8版のMecabの辞書が入っていますが、たぶんOpen JTalk用にカスタマイズされているでしょうから通常手順で導入したMecabの辞書を使うと何か問題があるかもしれません。

MMDAgent\_Example-1.4/Voice/mei ディレクトリにはMeiさんの他の音声情報もあるので-m オプションで指定する \*.htsvoice ファイルをいろいろ変えて試してみてください。

Meiさん音声

[MMDエージェント同梱のmei\\_normal.htsvoiceで合成](#)

nitech defaultさん音声

[Open Jtalkで配布している音声情報のnitech\\_jp\\_atr503\\_m001.htsvoiceで合成](#)

[音声合成, FreeBSD, Open JTalk, 技術資料](#)

From:

<https://wiki.hgotoh.jp/> - 努力したWiki

Permanent link:

<https://wiki.hgotoh.jp/documents/frebsd/openjtalk/openjtalk-001>

Last update: **2024/11/01 16:25**

