

# AquesTalk1 Linux プログラミングガイド

2020/12/01 アクエスト

<http://www.a-quest.com/>

## 1. 概要

本文書は、規則音声合成ライブラリ AquesTalk1 Linux をアプリケーションに組み込んで使用するためのプログラミングに関して、方法および注意点を示したものです。

AquesTalk1 Linux は Win 版の AquesTalk1 とは異なり、DA(サウンド出力)機能はありません。本ライブラリを用いて生成した音声波形データは、ファイルに出力、サウンドデバイスへ出力、またはネットワークを通じて転送するなど、アプリケーションの要求に応じた処理を別途実装する必要があります。

音声記号列の文字コードは、シフト JIS、EUC、UTF-8、UTF-16(LE)、ローマ字の各種を利用できます。それぞれ異なる名前の関数インターフェースが定義されています。

なお、本ライブラリをアプリケーションに組み込んで使用する際には、事前に「ソフトウェア使用許諾契約書」(aqt1\_lnx\_sdk\_lic.pdf)をご確認ください。

## 2. ライブラリ

AquesTalk1 Linux のライブラリは共有ライブラリ(.so)となっています。リンク時、および実行時に本ライブラリが必要になります。

声種、64/32bit 毎に libAquesTalk.so という同名の異なるファイルありますので、使用する声種、動作環境に応じて選択して用いてください。動的に声種を変更する場合は、libAquesTalk.so 共有ライブラリを、動的にロードしてください。

## 3. コンパイル・リンク

本ライブラリを使用する場合、ライブラリを呼び出すソースコードに AquesTalk1 Linux のヘッダファイル(AquesTalk.h)をインクルードしてコンパイルします。

リンク時には、-lAquesTalk を指定してリンクしてください。

## 4. 関数 API

### AquesTalk\_Synthe

AquesTalk.h

<b>説明</b>	音声記号列から音声波形を生成します 生成した音声データは、使用后、呼び出し側で AquesTalk_FreeWave を呼び出して開放してください。
<b>構文</b>	unsigned char * <b>AquesTalk_Synthe</b> (const char *koe, int iSpeed, int *size)
<b>引数</b>	
<i>koe</i>	音声記号列(NULL 終端 <b>シフト JIS</b> )
<i>iSpeed</i>	発話速度[%] 50-300 の間で指定 デフォルト: 100 値を大きく設定するほど、速くなる
<i>size</i>	生成した音声データのサイズが返る[byte](エラーの場合はエラーコードが返る)
<b>戻り値</b>	WAV フォーマットのデータ(内部で領域確保、解放は呼び出し側で AquesTalk_FreeWave()で行う)の先頭アドレスを返す。エラー時は、NULL を返す。このとき size にエラーコードが設定される。

### AquesTalk\_Synthe\_Euc

AquesTalk.h

---

説明	音声記号列から音声波形を生成します AquesTalk_Synthe()の EUC 文字コード版
構文	unsigned char * <b>AquesTalk_Synthe_Euc</b> (const char * <i>koe</i> , int <i>iSpeed</i> , int * <i>size</i> )
引数	
<i>koe</i>	音声記号列(NULL 終端 EUC) 以下、AquesTalk_Synthe()と同じ

---

### AquesTalk\_Synthe\_Utf8 AquesTalk.h

---

説明	音声記号列から音声波形を生成します AquesTalk_Synthe()の UTF-8 文字コード版
構文	unsigned char * <b>AquesTalk_Synthe_Utf8</b> (const char * <i>koe</i> , int <i>iSpeed</i> , int * <i>size</i> )
引数	
<i>koe</i>	音声記号列(NULL 終端 UTF-8 BOM は無し) 以下、AquesTalk_Synthe()と同じ

---

### AquesTalk\_Synthe\_Utf16 AquesTalk.h

---

説明	音声記号列から音声波形を生成します AquesTalk_Synthe()の UTF-16 文字コード版
構文	unsigned char * <b>AquesTalk_Synthe_Utf16</b> (const unsigned short * <i>wkoe</i> , int <i>iSpeed</i> , int * <i>size</i> )
引数	
<i>wkoe</i>	音声記号列(NULL 終端 UTF-16 BOM 指定は任意 エンディアンは実行環境に依存) 以下、AquesTalk_Synthe()と同じ

---

### AquesTalk\_Synthe\_Roman AquesTalk.h

---

説明	音声記号列から音声波形を生成します AquesTalk_Synthe()のローマ字(7bitASCII)文字コード版
構文	unsigned char * <b>AquesTalk_Synthe_Roman</b> (const char * <i>koe</i> , int <i>iSpeed</i> , int * <i>size</i> )
引数	
<i>koe</i>	音声記号列(NULL 終端 ASCII 表記方法はホームページ上の音声記号列仕様を参照) 以下、AquesTalk_Synthe()と同じ

---

### SetSpeed AquesTalk\_FreeWave AquesTalk.h

---

説明	音声データの領域を開放
構文	void <b>AquesTalk_FreeWave</b> (unsigned char * <i>wav</i> )
引数	なし
<i>wav</i>	WAV フォーマットのデータ(AquesTalk_Synthe()で生成した音声データ)
戻り値	なし

## 5. 音声データ形式

本ライブラリで生成する音声データは、次の形式となります。

AquesTalk\_Synthe()等で生成する音声データには、先頭部に WAV ヘッダが付与されています。

ストレート PCM データが必要な場合は、別途ヘッダを除いて使用してください。

ほかのフォーマットをご希望の場合はカスタマイズを承りますので、お問い合わせください。

サンプリング周波数	8KHz
量子化 bit 数	16bit
チャンネル数	モノラル
エンコード	リニア PCM
フォーマット	WAV 形式

## 6. 音声記号列

音声記号列の書き方、仕様については、

<http://www.a-quest.com/download/> の「音声記号列仕様」を参照ください。

## 7. サンプルコード

次に示すコードは、標準入力から音声記号列を1行読み込み、標準出力に音声波形(.wav フォーマット)を出力するプログラムです。

SampleTalk.c

```
#include <stdio.h>
#include "AquesTalk.h" // AquesTalk クラスのヘッダ

int main(int ac, char **av)
{
    int    size;
    int    iret;
    char   str[1024];

    // 音声記号列を入力
    if(fgets(str, 1024-1, stdin)==0) return 0;

    // 音声合成
    unsigned char *wav = AquesTalk_Synthe_Utf8(str, 100, &size);
    if(wav==0){
        fprintf(stderr, "ERR:%d¥n",size);
        return -1;
    }

    // 音声データ(wav フォーマット)の出力
    fwrite(wav, 1, size, stdout);

    // 音声データバッファの開放
    AquesTalk_FreeWave(wav);

    return 0;
}
```

サンプルコードのビルド方法を次に示します。

```
$ g++ -o SampleTalk SampleTalk.c -L./lib64/f1 -lAquesTalk
```

サンプルコードの実行方法を次に示します。

```
$ echo "これわ、ごーせー/お'んせーです。" | LD_LIBRARY_PATH=./lib64/f1 ./SampleTalk > test.wav
```

音声記号列にシェルの特許文字が含まれているので、ダブルクォーテーション(")で囲むなど、適当にエスケープする必要があります。

声種を変更する場合は、f1 の部分を他の声種に変更してください。

デフォルトの文字コードは UTF8 になっています。文字コードを変更する場合は、 SampleTalk.c の AquesTalk\_Synthe\_Utf8() 部分を文字コードに応じて変更してください。

test.wav に音声データが生成できれば OK です。エラーの場合には、文字コードやライブラリの配置が正しく行われているか再確認してください。

## 8. エラーコード表

関数が返すエラーコードの内容は、次の通りです。

値	内容
100	その他のエラー
101	メモリ不足
102	音声記号列に未定義の読み記号が指定された
103	韻律データの時間長がマイナスになっている
104	内部エラー(未定義の区切りコード検出)
105	音声記号列に未定義の読み記号が指定された
106	音声記号列のタグの指定が正しくない
107	タグの長さが制限を越えている(または[>]が見つからない)
108	タグ内の値の指定が正しくない
109	WAVE 再生ができない(サウンドドライバ関連の問題)
110	WAVE 再生ができない(サウンドドライバ関連の問題 非同期再生)
111	発声すべきデータがない
200	音声記号列が長すぎる
201	1つのフレーズ中の読み記号が多すぎる
202	音声記号列が長い(内部バッファオーバー1)
203	ヒープメモリ不足
204	音声記号列が長い(内部バッファオーバー1)
205	ライセンスキーが正しくない。または、設定されていない。
上記以外	音声記号列エラー(音声記号列上でエラーの位置を返す)

## 9. 履歴

日付	版	変更箇所	更新内容	更新者
2007/1/7	1.0	新規作成	Win 版から加筆修正	N. Y
2007/9/27	2.2	Ver. up	各種文字コード IF 追加	N. Y
2010/1/2	2.3	Ver. up	ライセンスキー不要	N. Y
2020/12/01	1.7	Ver. up	多声種化、バグフィックス	N. Y