

# 偽装ちゃん22vの説明

2025-10-03

一番最初の確認を追記

2024-08-28

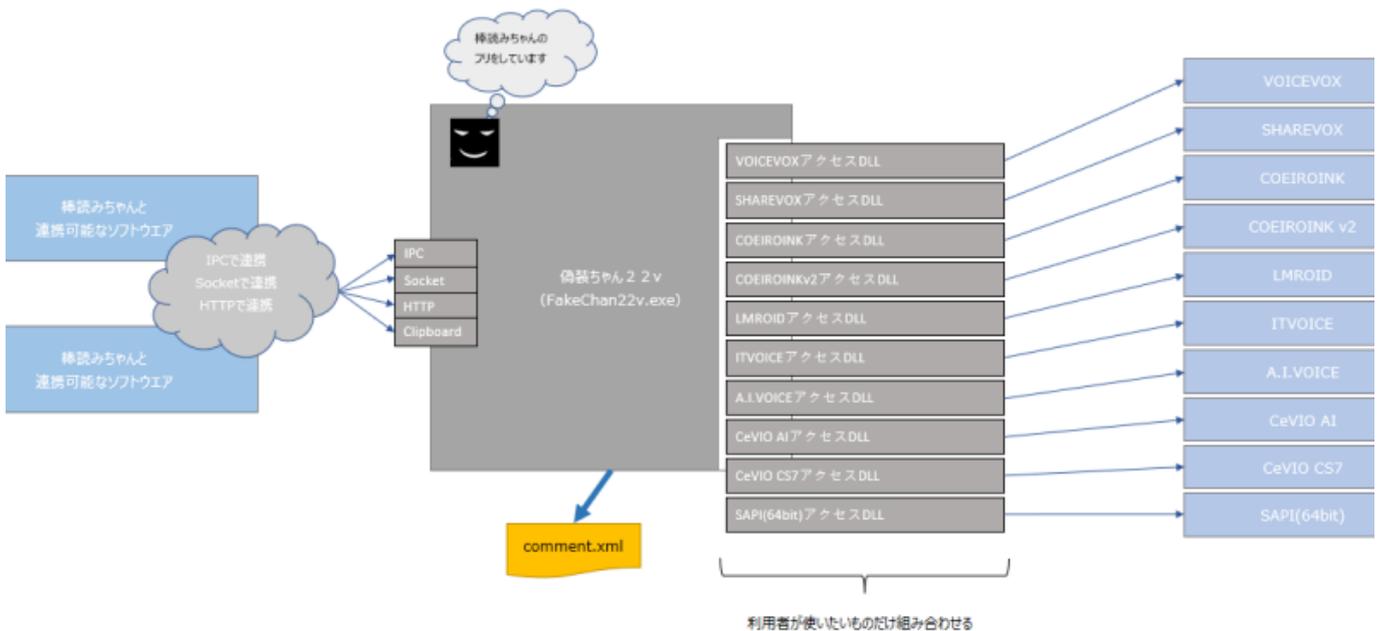
使い方を追記

## これは何？

「偽装ちゃん22v」です。

棒読みちゃんの替わりに起動すると、コメントビューなどからのメッセージを受け取って音声合成製品に発声させます。

「偽装ちゃん22」から「AssistantSeikaとの接続機能を省き、利用者が自身で利用製品(音声合成製品アクセスDLL)を選択できるようにしています。



## どうしてこれを作ったの？

- 「音声合成製品と AssistantSeika と 偽装ちゃん22 と コメントビュー を起動しなくちゃいけないので面倒、立ち上げるもの多過ぎ」という方がいるのは認識しています。特に「AssistantSeika と 偽装ちゃん22」の2つが必要な事に気が入らない人が結構いるようです。AssistantSeikaを削ったところで起動するものが多いのは変わらないと思うのですが、不要なら使う的な事を言われたりしましたので作成しました。
- 「自分が使わない製品のコードを含んだものを使いたくない」という事を言われてしまいましたので、利用者自身で必要な機能を含んだモジュール(DLL)を組み合わせ利用してもらおう方式に改めました。

# ダウンロード



こちらで公開しました。  
<https://k896951.booth.pm/items/5065157>

## バリエーション

バリエーション	説明		
AssistantSeika専用版	AssistantSeikaと接続する製品アクセスDLLとの組み合わせです。偽装ちゃん2-2とほぼ同じ動作をします。AssistantSeikaがサポートする音声合成製品を利用できます。		
SAPI版	以下の音声合成製品アクセスDLLを同梱しています。 SAPI(64bit)		
AIコレクション版	以下の音声合成製品アクセスDLLを同梱しています。 A.I.VOICE CeVIO/AI SAPI(64bit)		
TTSコレクション版	以下の音声合成製品アクセスDLLを同梱しています。 VOICEVOX SHAREVOX COEIROINK COEIROINKv2 LMDROID ITVOICE SAPI(64bit)		
音声合成製品アクセスDLL	<b>音声合成製品アクセスDLL単品です。対応製品を増やしたい場合にお使いください。</b>		
	#1	SAPI(64bit)を利用する為の製品アクセスDLLです。SAPI版、AIコレクション版、TTSコレクション版、の「偽装ちゃん」に同梱しています。	FakeChan22BackendSAPI64.dll
	#2	A.I.VOICEを利用する為の製品アクセスDLLです。	FakeChan22BackendAIVOICE.dll
	#3	CeVIO AIを利用する為の製品アクセスDLLです。	FakeChan22BackendCEVIOAI.dll
	#4	CeVIO CS7を利用する為の製品アクセスDLLです。	FakeChan22BackendCEVIOCS7.dll
	#5	VOICEVOXを利用する為の製品アクセスDLLです。	FakeChan22BackendVOICEVOX.dll
	#6	COEIROINKを利用する為の製品アクセスDLLです。	FakeChan22BackendCOEIROINK.dll
	#7	COEIROINK v2を利用する為の製品アクセスDLLです。	FakeChan22BackendCOEIROINKV2.dll
	#8	SHAREVOXを利用する為の製品アクセスDLLです。	FakeChan22BackendSHAREVOX.dll
	#9	LMROIDを利用する為の製品アクセスDLLです。	FakeChan22BackendLMROID.dll
	#10	ITVOICEを利用する為の製品アクセスDLLです。	FakeChan22BackendITVOICE.dll
	#11	VOICEVOX Nemoを利用する為の製品アクセスDLLです。	FakeChan22BackendVOICEVOXNemo.dll
#12	AssistantSeikaを利用する為の製品アクセスDLLです。	FakeChan22BackendAssistantSeika.dll	
リスナ拡張DLL	<b>機能拡張DLL単品です。必要な機能があればお求めください。</b>		
	拡張ファイル監視	テキストファイルへの行追加を検出し差分を読み上げるためのDLLです。ログファイルの読み上げを想定しています。	TaskExtendFileWatcher.dll
	拡張Httpリスナ	IPアドレス・ポート番号指定が可能なHTTP受付用DLLです。ポート変更が必要な場合や、外部PCから接続が必要な場合はこちらを利用できます。 外部からHTTPで「偽装ちゃん」を実行しているPCへ接続させる場合は「偽装ちゃん」を管理者権限で起動する必要があります。	TaskExtendHttpWatcher.dll
	拡張Socketリスナ	IPアドレス・ポート番号指定が可能なSocket受付用DLLです。ポート変更が必要な場合や、外部PCから接続が必要な場合はこちらを利用できます。 外部からSocketで「偽装ちゃん」を実行しているPCへ接続させる場合は「偽装ちゃん」を管理者権限で起動する必要があります。	TaskExtendSocketWatcher.dll
WebSocket受付リスナ	<a href="https://github.com/chocoo/BouyomiChan-WebSocket-Plugin">https://github.com/chocoo/BouyomiChan-WebSocket-Plugin</a> で公開されているWebSocket受付用プラグインのプロトコルを受付出来るDLLです。 17LIVE用コメントビューア17Liveコメントビューア-47との連携で使われている模様です。 外部からHTTPで「偽装ちゃん」を実行しているPCへ接続させる場合は「偽装ちゃん」を管理者権限で起動する必要があります。	TaskExtendWebSocketWatcher.dll	

「偽装ちゃん」で利用可能な音声合成製品アクセスDLL、リスナ拡張DLLを別途BOOTHで公開しています。  
 購入 or ダウンロードしたzipファイルから製品アクセスDLLを取り出してフォルダExtendにコピーして

「偽装ちゃん□□□□を起動してください。

## 注意事項

- 「偽装ちゃん□□□□で扱うcidはAssistantSeikaのcidと互換性がありません。
- #12がコピーされていると、この音声合成製品アクセスDLLのみ有効となります。他のDLLは利用されません。
- CeVIO/AI, CeVIO/CS7仕様上の制限で、他のアプリケーションと同時に「偽装ちゃん□□□□を利用する事ができないかもしれません。

## 使用しているサードパーティ製品

- DLLをまとめる目的で Costura.Fody を利用しています。

```
バージョン 5.7.0
作成者      geertvanhorrik, simoncropp
ライセンス  https://licenses.nuget.org/MIT
```

- Costura.Fody の依存関係により Fody を利用しています。

```
バージョン 6.8.1
作成者      Fody
ライセンス  https://www.nuget.org/packages/Fody/6.8.1/license
```

- 音声再生のため NAudio を利用しています。

```
NAudio.Core    2.2.1    Mark Heath    MIT
NAudio.Wasapi  2.2.1    Mark Heath    MIT
```

## 使い方

### 一番最初の確認

連携ソフトウェアから偽装ちゃん22vが呼び出される時、連携ソフトウェアは偽装ちゃん22vを「棒読みちゃん」と思い込んで動作します。

なので、棒読みちゃんと連携できるPC環境でなければ当然、偽装ちゃん22vとも連携できません。

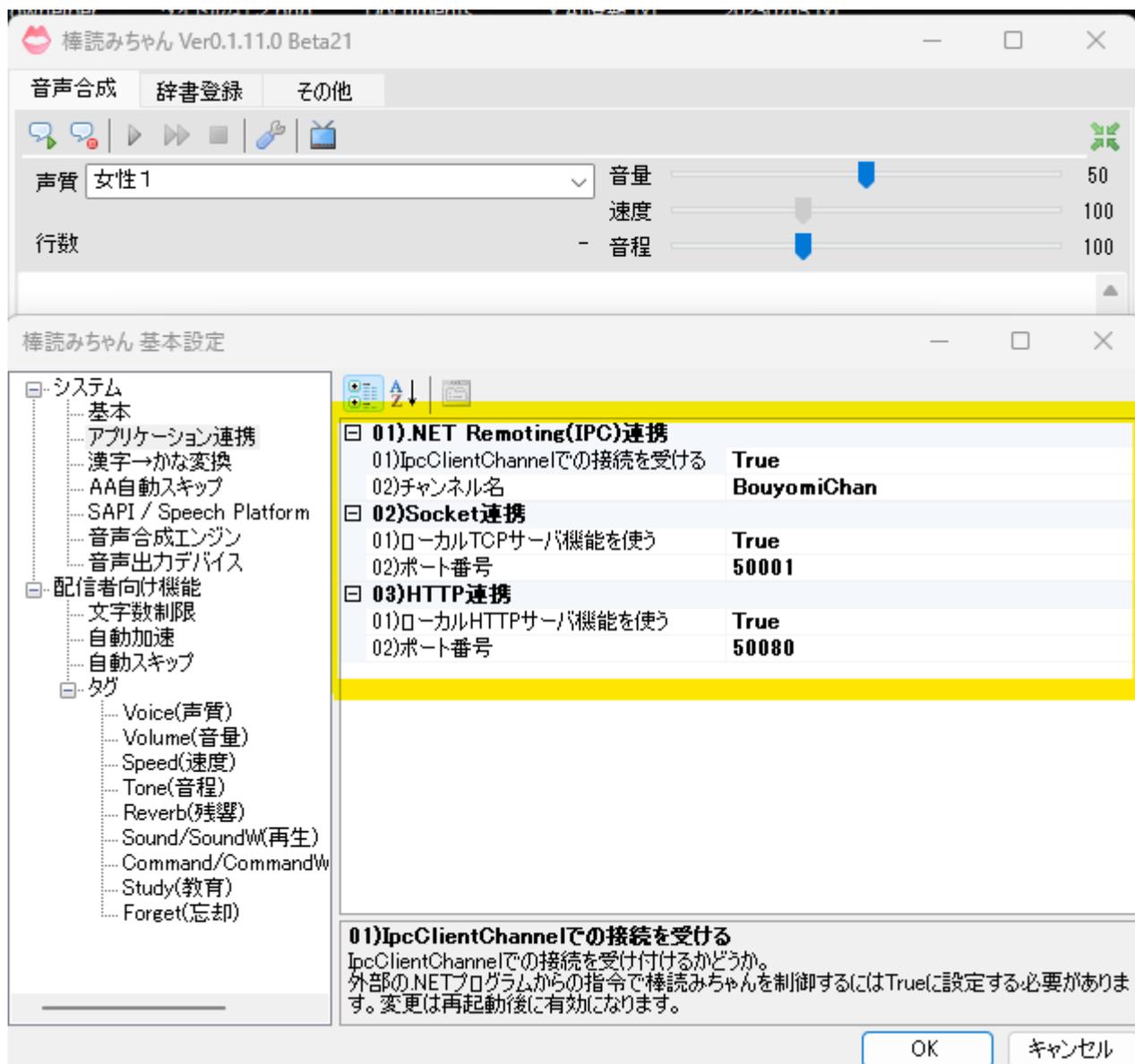
最初に棒読みちゃんが使えるか確認しましょう。 確認を取ってから偽装ちゃん22vを入手しても遅くはありません

連携ソフトウェアから棒読みちゃんが発声できるようであれば、次に連携ソフトウェアが採用している連携方式の確認をします。

偽装ちゃん製作者の手元にある棒読みちゃんは画像にあるように3種類の連携方式があります。

\*WebSocketの連携は専用DLLが提供されているのでそれを追加してください。

自分が使いたいソフトウェアの連携方法を確認したら、棒読みちゃんを終了し、偽装ちゃん22vを起動して対応するリスナを有効にしましょう。

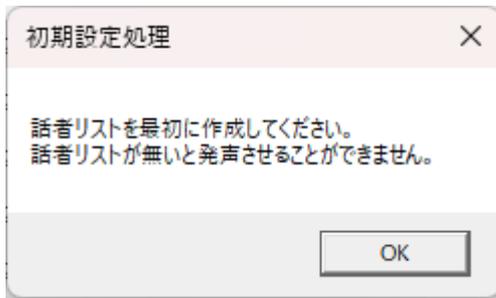


棒読みちゃんの連携方式	対応する偽装ちゃん22vのリスナ	連携ソフトウェアの例
.NET Remoting (IPC)連携	棒読みちゃんIPC接続 BouyomiChan/Remoting	MultiCommentViewer
Socket連携	Socketリスナ 127:0.0.1:50001	ゆかりねっと
HTTP連携	HTTPリスナ 127:0.0.1:50080	わんコメ

棒読みちゃんの連携方式でSocketやHTTPが機能しない場合、アンチウイルス製品やOSの管理によって利用できない場合があります。  
 OSの管理によるものであった場合は[管理TCPポート除外設定ユーティリティ](#)で利用可能に出来るかもしれません。ただしこれの使い方はリンク先にある以上の説明はしません。理解できそうになれば残念ながら偽装ちゃん22vの利用を諦めて別の手段を探したほうが良いと思います。

## 初回

初めて FakeChan22v.exe ( BouyomiChan.exeにリネームしている場合はBouyomiChan.exe ) を立ち上げるとこの表示が行われます。

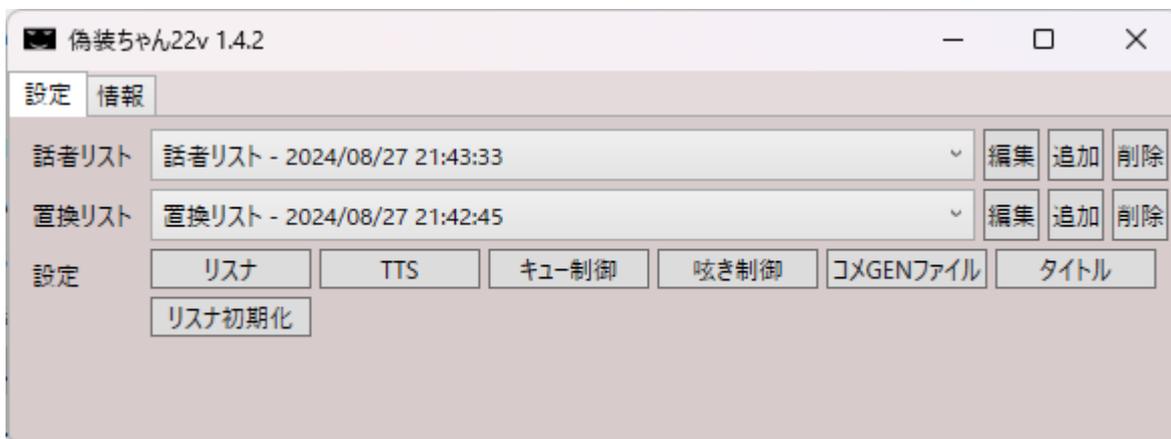


OKボタンを押すと話者リストのエディタへ遷移します。

話者リストに話者を登録し保存ボタンで保存を行うと、メインウィンドウへ遷移します。

## メインウィンドウ

2回目起動以降、最初に表示されるウィンドウです。



このウィンドウから、話者リストの編集、置換リストの編集、リスナ設定編集(TTS(音声合成製品)設定編集、キュー制御編集、呟き制御編集、コメントジェネレータファイル設定、ウィンドウタイトル設定、リスナ初期化、を行います。

## 話者リスト作成/編集

初回起動時に遷移させられる、話者リストのエディタです。2回目以降はメインウィンドウにある話者リストの追加、編集のボタンを押して表示されます。

話者リストは、目的に従って話者を集約したリストになります。例えば音声製品毎の話者群であったり、女性 男性だけの話者群であったり、英語話者を集めたものであったり、目的によって複数作成できます。



話者一覧から話者を選択し、追加ボタンを押すと話者リストに話者が追加されます。フィルタで話者名を絞り込む事もできます。

全部追加ボタンを押すと話者一覧の全話者を話者リストに追加します。並びの変更は、話者リストの対象話者を選択してから 移動ボタン、 移動ボタンを使ってください。話者リストの話者を選択して削除ボタンを押すと削除されます。利用のチェックを外すと、話者リストにあっても利用されない話者となります。同じ話者を複数登録しても構いません。

話者リストの話者名は修正が可能です。必要なら音声パラメタを調整し、パラメタに対応した名称にすると分かりやすいかもしれません。話者リストの識別には、テキスト先頭に記述する話者指定の1~3桁英数字文字列を設定します。すべて設定する必要はありません。識別は大文字小文字を区別します。また同一の識別を指定した場合は後勝ちです。話者リストの先頭から辿って最後に設定された話者に適用されます。

テキスト	選択される話者
"T2)こんばんは！"	cid=70000 さとうささら
"T1)こんばんは！"	cid=70004 弦巻マキ (日)
"t2)こんばんは！"	cid=70008 すずきつづみ

保存ボタンで話者リストに話者リスト名を付けて保存します。目的や用途が明確な名前に変更してください。同じ話者リスト名で話者リストを複数登録できますが、管理が面倒になるのでお勧めしません。

## 置換リスト定義編集

音声合成製品に発声させるテキストの置換を行うための定義を行うエディタです。メインウインドウにある定義リストの追加,編集のボタンを押して表示されます。

置換リスト定義編集ウインドウ左端の一覧は、処理と処理の順序を示しています。一覧の先頭から順番に「実行」にチェックの入った処理を実行します。一覧の上部にある移動ボタンで選択した処理の順序を変更する事が出来ます。

変換テストを使って変換処理の結果を確認できます。

置換リスト定義名にはデフォルトの名称が設定されています。目的や用途が明確な名前に変更してください。同じ置換リスト定義名で置換リスト定義を複数登録できますが、管理が面倒になるのでお勧めしません。

## 話者指定検出

実行	処理名	[処理の説明]
<input checked="" type="checkbox"/>	話者指定検出	適用
<input type="checkbox"/>	URL置換	補正
<input checked="" type="checkbox"/>	w変換	
<input checked="" type="checkbox"/>	8変換	
<input type="checkbox"/>	絵文字置換	
<input type="checkbox"/>	絵文字除去	
<input type="checkbox"/>	全角英字記号置換	
<input type="checkbox"/>	全角数字置換	
<input checked="" type="checkbox"/>	テキスト置換	
<input checked="" type="checkbox"/>	切捨て	

[処理の説明] テキスト先頭の話者指定を分離します。OFFだと話者指定が有効になりません

適用  分離した話者指定を有効にします

補正  全角記述の話者指定に対応します

変換テスト

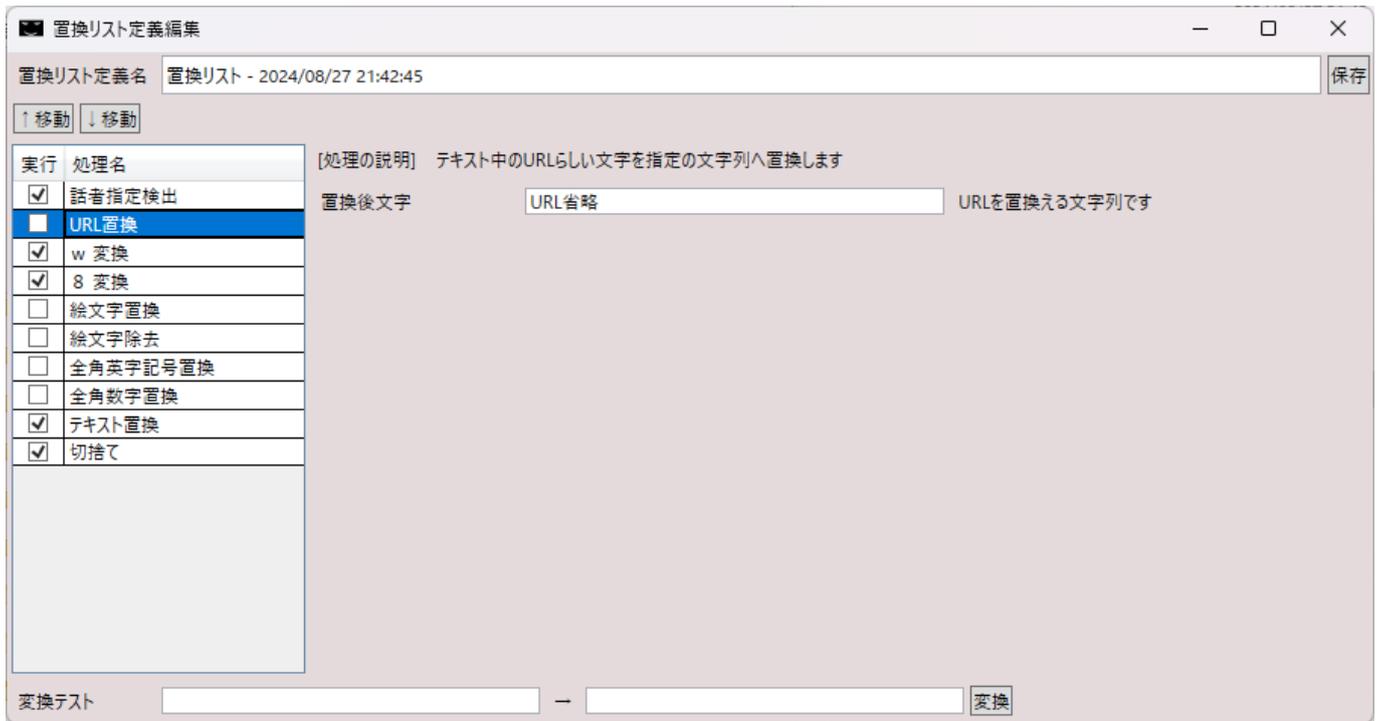
「実行」にチェックが入り、「適用」にチェックが入っていると、テキスト先頭の話者指定記述（全て半角の文字）を検出して使用するようになります。例えば

### y5) コンバンワ

は“y5)”と“コンバンワ”に分離され、話者リストの“y5)”と紐づけられた話者で“コンバンワ”を発声させることになります。

「補正」にチェックが入っていると、話者指定の全角記述を許容するようになります。

## URL置換



「実行」にチェックが入っていると、テキスト中のURLと推測される個所を指定の文字列で置換します。この例だと、

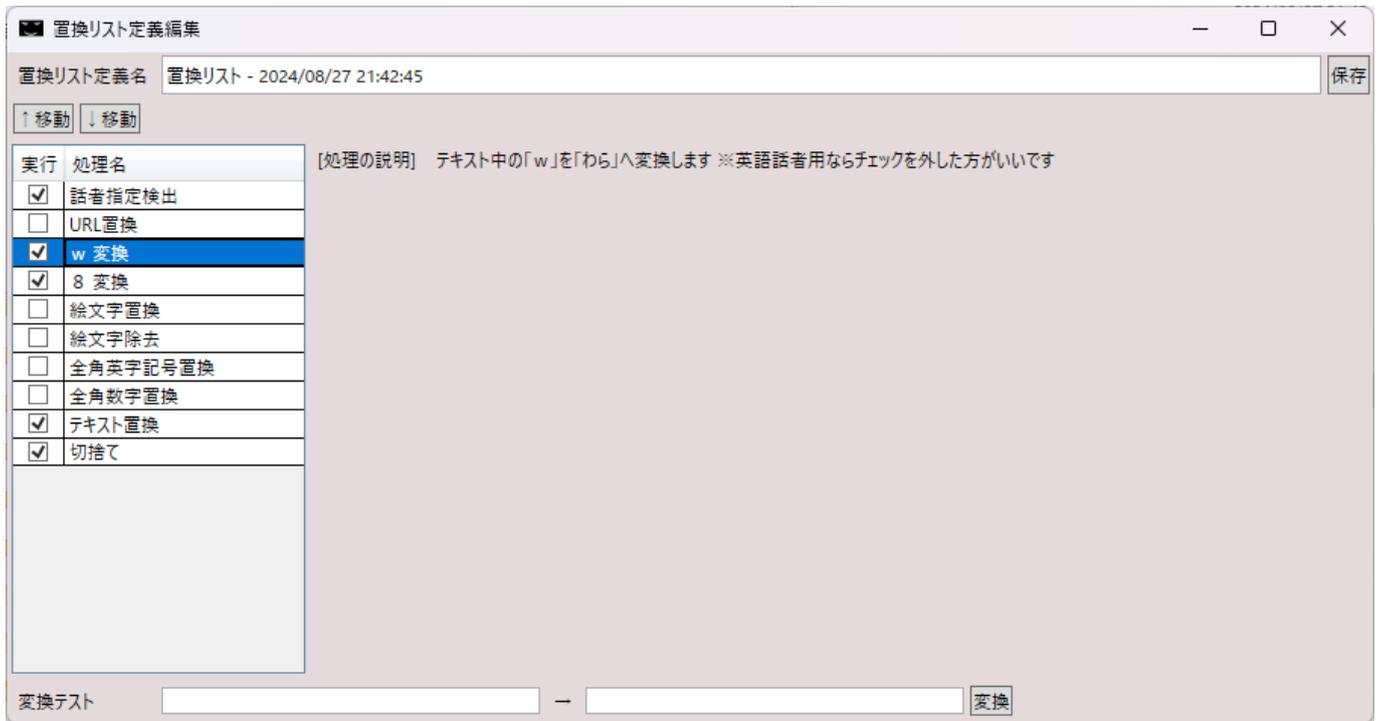
検索エンジンはこちら <https://www.google.com>

は

検索エンジンはこちら URL省略

と変換されます。

□変換



置換リスト定義編集

置換リスト定義名 置換リスト - 2024/08/27 21:42:45 保存

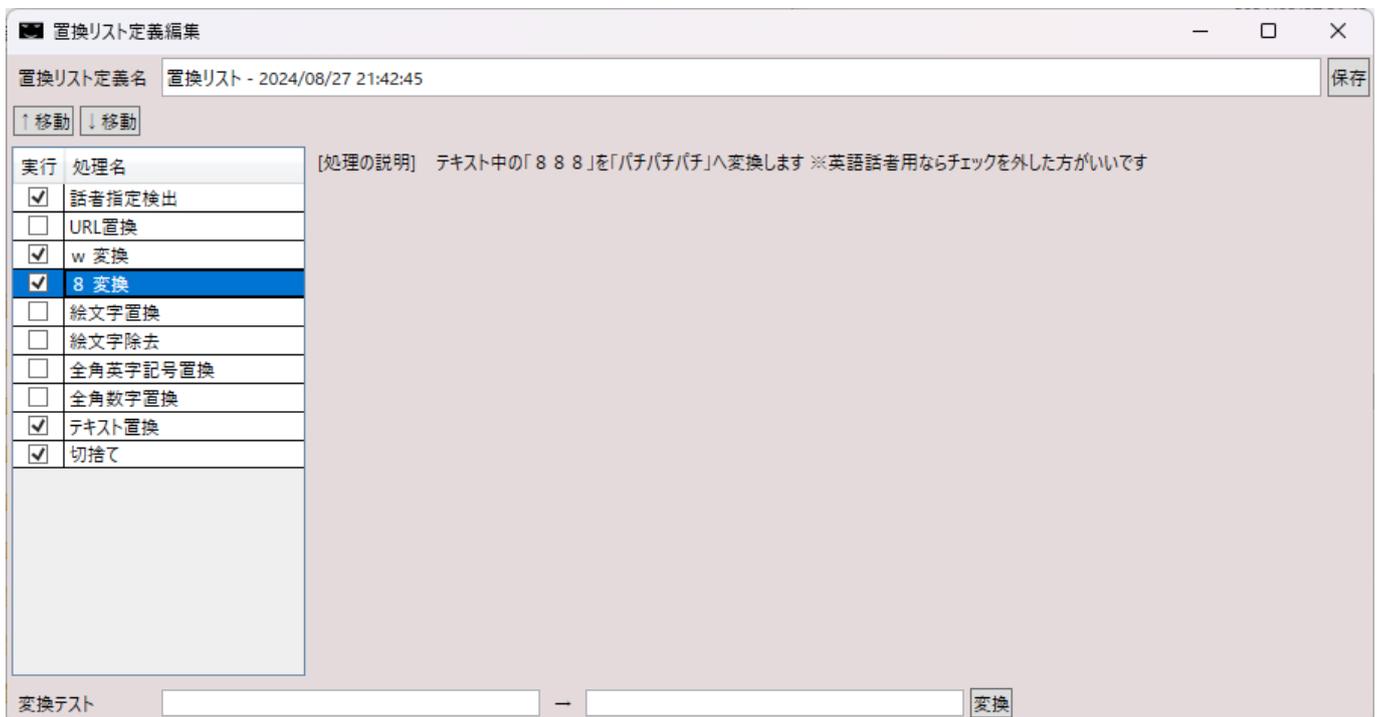
↑移動 ↓移動

実行	処理名	[処理の説明]
<input checked="" type="checkbox"/>	話者指定検出	テキスト中の「w」を「わら」へ変換します ※英語話者用ならチェックを外した方がいいです
<input type="checkbox"/>	URL置換	
<input checked="" type="checkbox"/>	w 変換	
<input checked="" type="checkbox"/>	8 変換	
<input type="checkbox"/>	絵文字置換	
<input type="checkbox"/>	絵文字除去	
<input type="checkbox"/>	全角英字記号置換	
<input type="checkbox"/>	全角数字置換	
<input checked="" type="checkbox"/>	テキスト置換	
<input checked="" type="checkbox"/>	切捨て	

変換テスト  -  変換

「実行」にチェックが入っていると、テキスト中の連続した□□□や、おそらく「笑」の意であろう□□□を「わら」に変換します。

## 8 変換



置換リスト定義編集

置換リスト定義名 置換リスト - 2024/08/27 21:42:45 保存

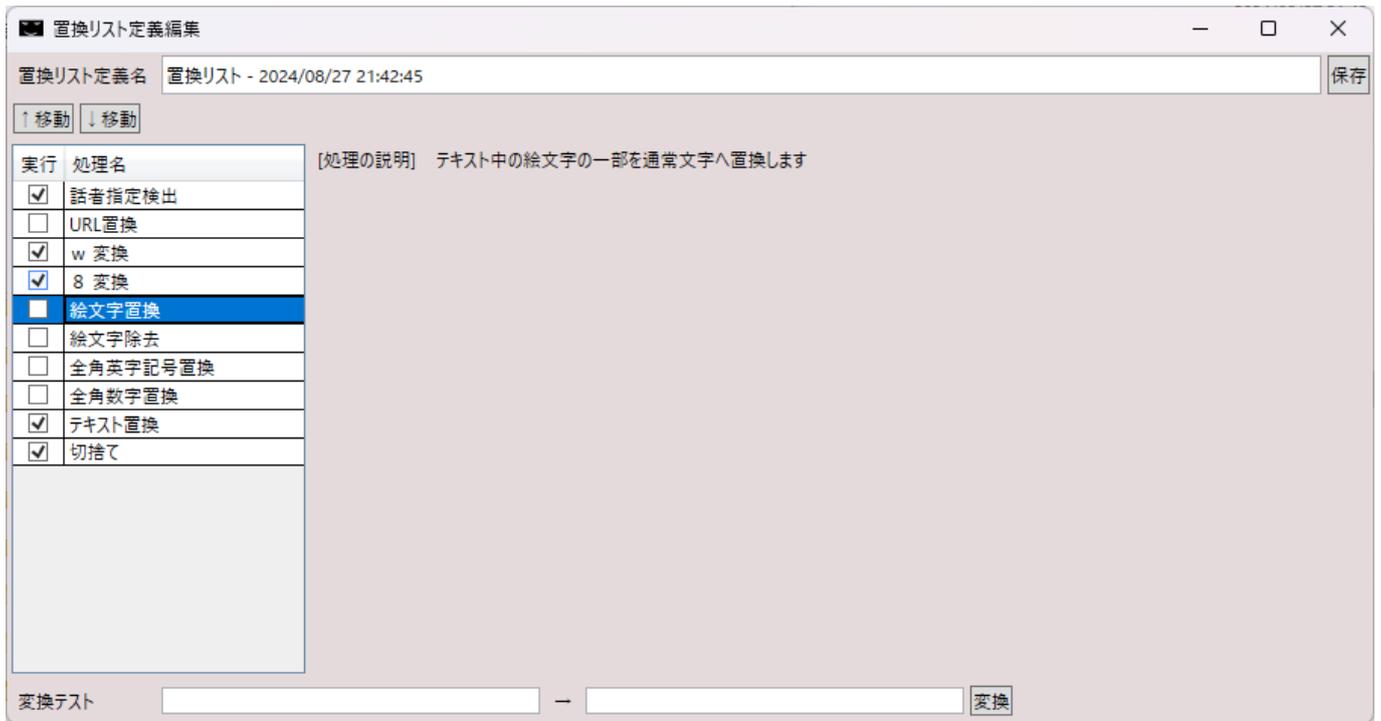
↑移動 ↓移動

実行	処理名	[処理の説明]
<input checked="" type="checkbox"/>	話者指定検出	テキスト中の「8 8 8」を「パチパチパチ」へ変換します ※英語話者用ならチェックを外した方がいいです
<input type="checkbox"/>	URL置換	
<input checked="" type="checkbox"/>	w 変換	
<input checked="" type="checkbox"/>	8 変換	
<input type="checkbox"/>	絵文字置換	
<input type="checkbox"/>	絵文字除去	
<input type="checkbox"/>	全角英字記号置換	
<input type="checkbox"/>	全角数字置換	
<input checked="" type="checkbox"/>	テキスト置換	
<input checked="" type="checkbox"/>	切捨て	

変換テスト  -  変換

「実行」にチェックが入っていると、テキスト中の連続した「8」を「パチ」に変換します。

## 絵文字置換



「実行」にチェックが入っていると、テキスト中の絵文字で可能な場合は通常テキスト表記に変換します。例えば

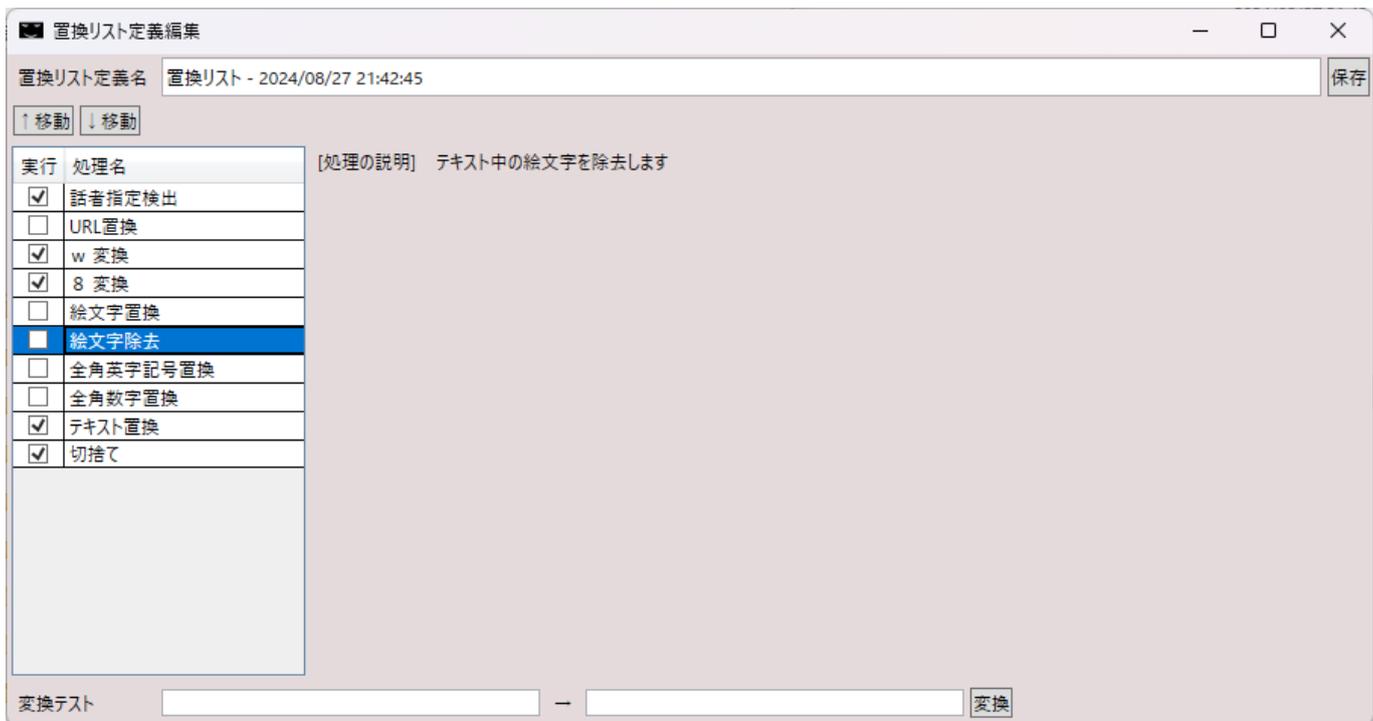
開店時間は です！

は

開店時間は 10時半です！

と変換されます。

絵文字除去



「実行」にチェックが入っていると、テキスト中の絵文字を除去します。例えば

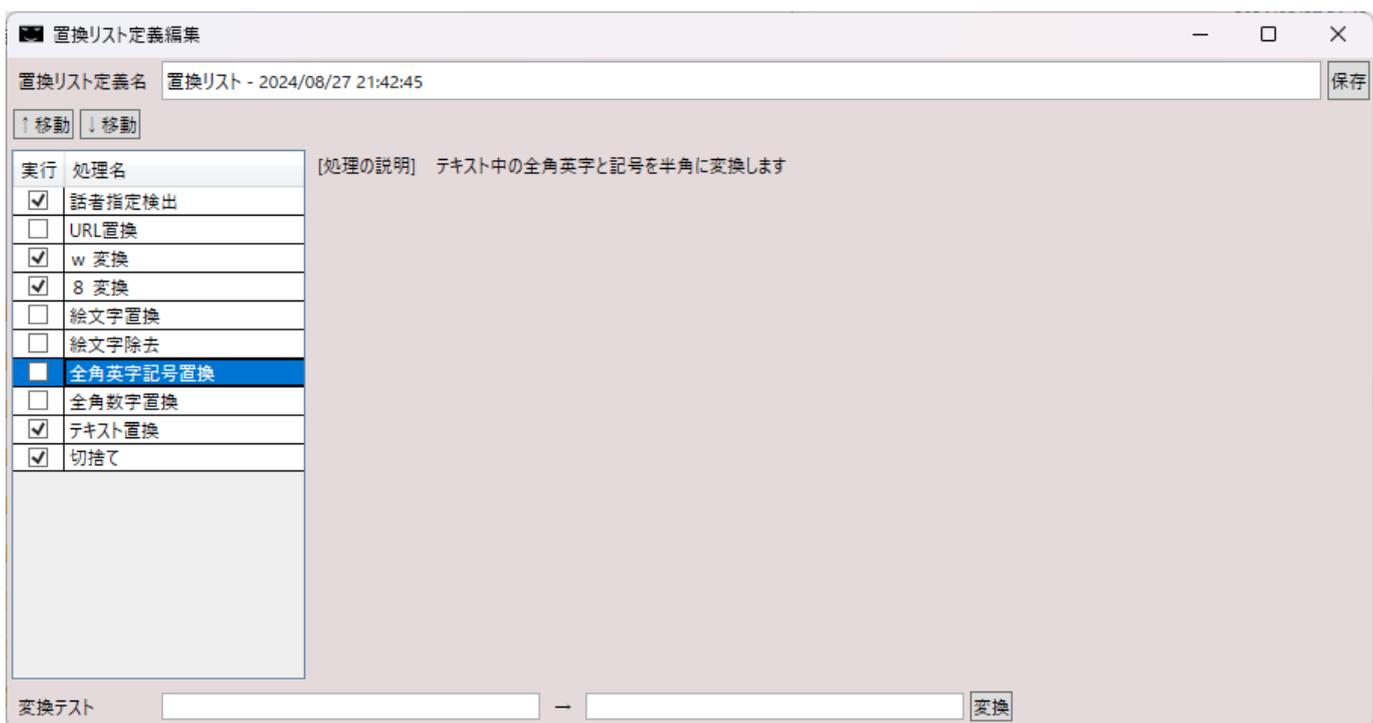
拍手～

は

拍手～

と変換されます。

### 全角英字記号変換



「実行」にチェックが入っていると、テキスト中の全角英字記号を半角に変換します。例えば

□□□□□□

は

YAHOO!?

と変換されます。

### 全角数字変換



「実行」にチェックが入っていると、テキスト中の全角数字を半角に変換します。例えば

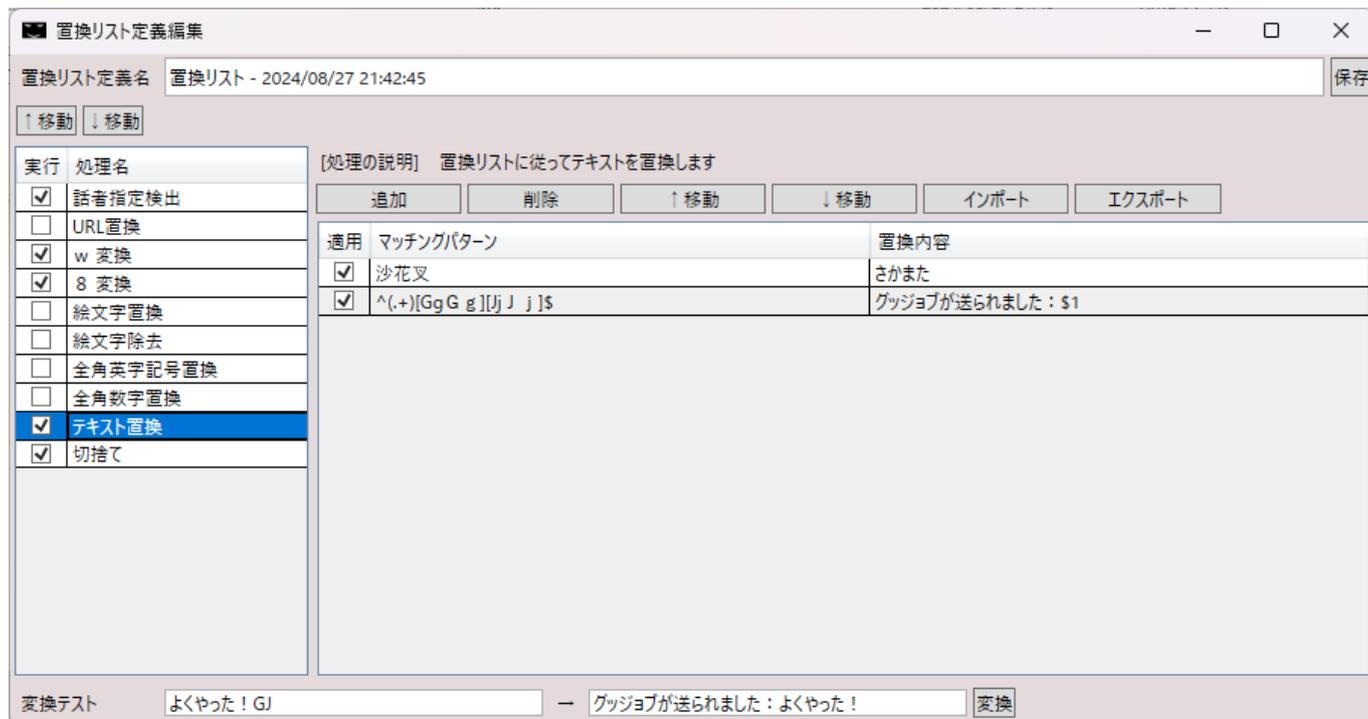
□□□□□□

は

3141592

と変換されます。アラビア数値のみの処理なので注意してください。

### テキスト置換



「実行」にチェックが入っていると、置換処理の定義に従ってテキストの置換を行います。追加ボタンを押すと置換定義が追加されます。並びの変更は、置換定義を選択してから 移動ボタン、 移動ボタンを使ってください。削除で選択した置換定義を削除します。適用のチェックを外すと置換定義リストにあっても利用されない置換定義となります。

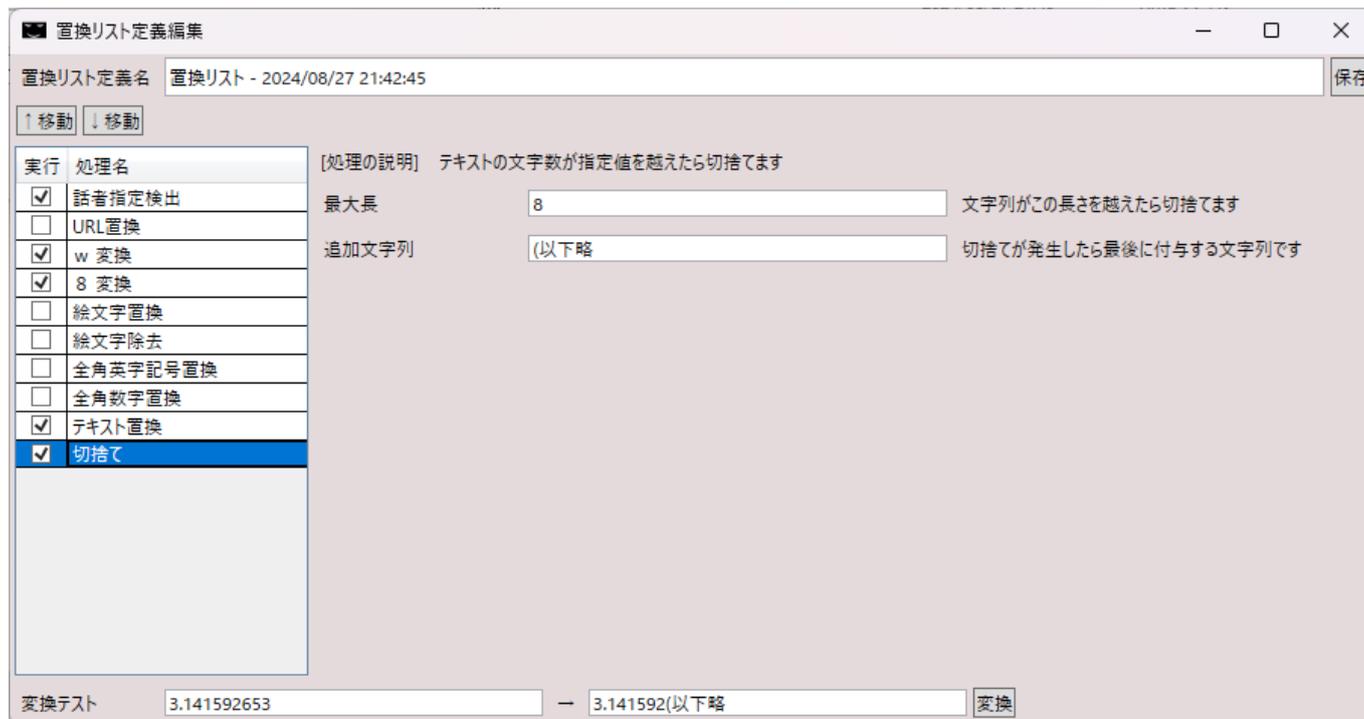
置換リストは目的によって複数作成できます。テキストの「ですます調」を「である調」にする置換、特定のライブ配信で補正する必要があるキーワードの置換、の様に複数作成できます。もちろん一つのリストにすべて登録しても構いません。

マッチングパターン、置換内容にはMicrosoft .NET Framework の正規表現を適用できます。

マッチングパターン	置換内容	置換前	置換結果
沙花叉	さかまた	沙花叉クロエ	さかまたクロエ
^(.+)[Gg][Jj]\$	グッジョブが送られました: \$1	よくやった GJ	グッジョブが送られました: よくやった!

名前の読みが難しそうな場合に登録して対応する等に利用できます。リストの先頭から順に適用していきます。

### 切捨て



「実行」にチェックが入っていると、先頭から「最大長」の文字数だけ選択して以降の文字を切捨てします。その後「追加文字列」で指定する文字列を追加します。例えば最大長を8とした場合

3.141592653

は

3.141592(以下略

と変換されます。

### 変換テスト

テキストを入力して一覧の各処理の処理結果を確認する事が出来ます。

### その他

- テキストの内容から話者を処理「テキスト置換」で指定する時は、処理「話者指定検出」よりも先に実行してください。処理「話者指定検出」の後に処理「テキスト置換」を使ってテキスト先頭に話者指定の文字を追加しても有効になりません。
- 非日本語話者で使用する場合、処理「全角英字記号変換」と処理「全角数字置換」を実行して全角文字を変換しておくことで発声失敗が少なくなります。
- A.I.VOICEやVOICEROID+EX、VOICEROID2では絵文字が含まれていると動作がおかしくなる場合があります。処理「絵文字除去」で絵文字を消し込んでおくとも良いかもしれません。
- 非日本語判定は、一覧の処理実行前に実施されています。この処理で変更する事はできません。
- 非日本語判定時に使用する置換リスト定義では、処理「変換」「8変換」を利用しないほうが良いです。「わら」「パチ」の文字が埋め込まれてしまうため発声に失敗する可能性があります。

## リスナインタフェース設定

偽装ちゃん22v がテキストの受信を行うインタフェースの設定を行うエディタです。メインウインドウにあるボタン「リスナ」を押して表示します。  
編集するインタフェースを選択してから設定を行ってください。

リスナ名は、メインウインドウで選択したインタフェース(リスナ)の名称です。

リスナ名	説明
棒読みちゃんIPC接続 BouyomiChan/Remoting	棒読みちゃんの IpcClientChannel接続を模しています。連携ソフトウェアがIPCを利用する場合はここを有効にします。
Socketリスナ 127.0.0.1:50001	棒読みちゃんの Socket接続ポート50001を模しています。連携ソフトウェアがSocketを利用する場合はここを有効にします。
Socketリスナ 127.0.0.1:50002	棒読みちゃんの Socket接続ポート50002を模しています。連携ソフトウェアがSocketを利用する場合はここを有効にします。
HTTPリスナ 127.0.0.1:50080	棒読みちゃんの HTTP接続ポート50080模しています。連携ソフトウェアがHTTPを利用する場合はここを有効にします。
HTTPリスナ 127.0.0.1:50081	棒読みちゃんの HTTP接続ポート50081模しています。連携ソフトウェアがHTTPを利用する場合はここを有効にします。
クリップボード監視	OSのクリップボードにコピーされたテキストを入力として利用する場合はここを有効にします。棒読みちゃんのクリップボード監視プラグインと類似の機能です。

- リスナ利用のチェックを入れて、保存ボタンを押すと、テキスト受信を開始します。
- 話者リスト（通常時）には、発声で使用する話者リストを指定します。
- 置換リスト（通常時）には、テキスト置換で使用する置換リストを指定します。
- 非日本語判定にチェックを入れて、保存ボタンを押すと、テキスト中の非日本語文字が指定割合以上含まれている場合に非日本語と判定されます。  
判定は置換処理の前に実施されます。
- 話者リスト（非日本語時）には、非日本語と判定が行われた時に発声で使用する話者リストを指定します。
- 置換リスト（非日本語時）には、非日本語と判定が行われた時にテキスト置換で使用する置換リストを指定します。
- 非同期の音声再生にチェックを入れて、保存ボタンを押すと、テキストをキューで順番管理せず

非同期に発声します。

- 話者のランダム選択にチェックを入れて、保存ボタンを押すと、話者リスト（通常時）で指定した話者リストの（利用にチェックが入った）話者の中からランダムに発声する話者を選択します。
- サービス名は、コメントジェネレータで生成する comment.xml のservice属性に設定する文字列です。

リスナ拡張DLLを追加すると、利用可能になったリスナが一覧に追加されます。

## クリップボード監視の取得無効期間

リスナ「クリップボード監視」には設定「クリップボード取得無効間隔」を設定してください。同じ内容のクリップボード更新通知が連続して偽装ちゃん22vに届いた場合、前回受信した通知と今回受信した通知の間隔が指定した時間以下の場合に無効として扱います。

利用者から「同じ内容を連続発声するのを抑止してほしい」とのリクエストがありました。偽装ちゃん22vはOSから届くクリップボード更新通知を素直に受信して発声処理を行っていました。ですが、その通知が

- 1回のクリップボードへのコピー操作で複数生じた通知の中の1つなのか
- 複数回同じ内容をクリップボードへのコピーする操作の結果の通知なのか

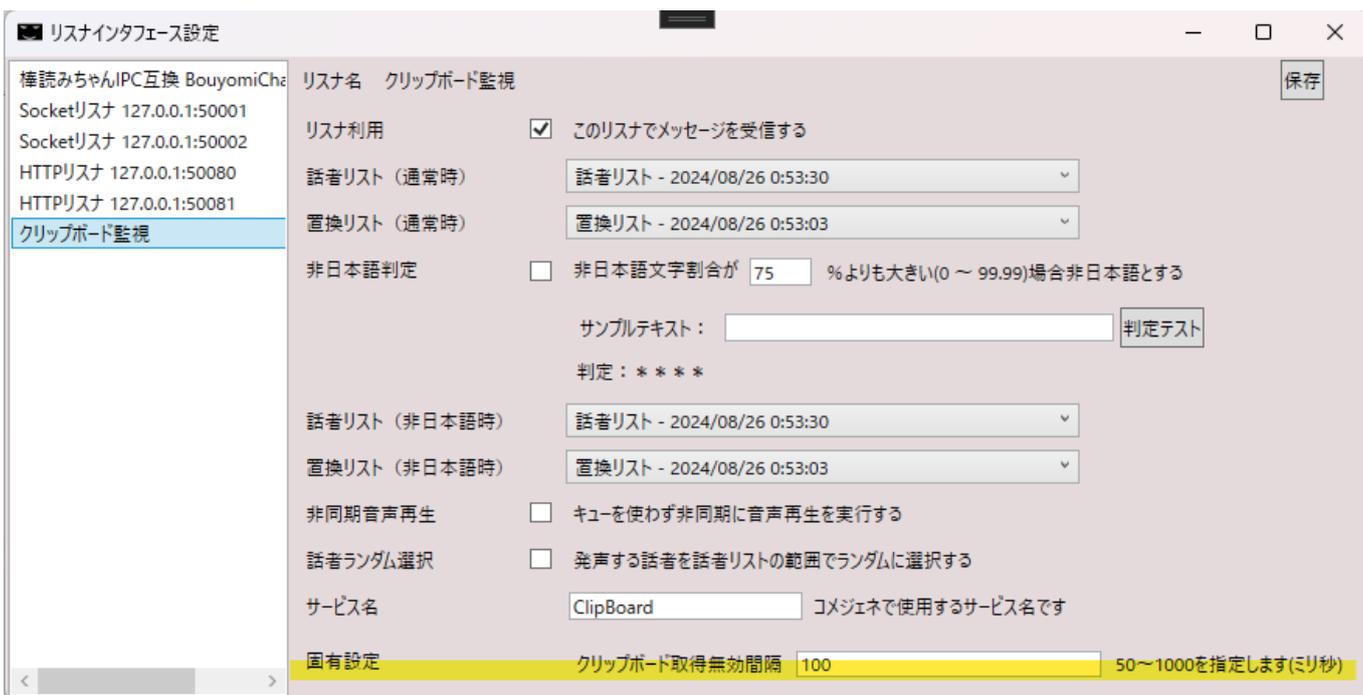
を判断できません。

偽装ちゃん22vで使用するフレームワークでは回避できない模様です

結果、偽装ちゃん22vの責ではないのですが利用者からは「偽装ちゃん22vがクリップボードのテキストを繰り返し発声させている」ように見えてしまいます。

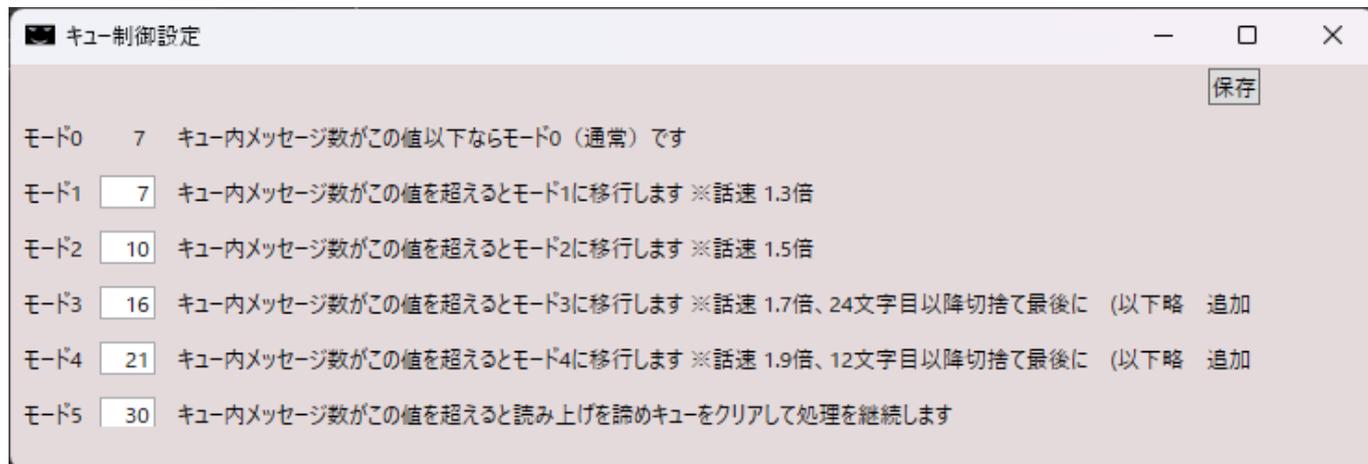
他のクリップボードを扱うアプリケーションの公開情報を調べると、前回の通知から100ms以下の間隔で届いた通知を無効とする実装をした事例があると分かりました。偽装ちゃん22vで試行してみると少なくとも開発者環境では回避できるようだったので、この手法を採用する事にしました。

利用するアプリケーションと実行環境に合わせて「クリップボード取得無効間隔」を調整してください。



## キュー制御設定

偽装ちゃん22が受信したテキスト（メッセージ）を登録したキューを制御するための境界値を設定するエディタです。メインウインドウにあるボタン「キュー制御」を押して表示します。



キューから取り出したテキストの扱いを決める境界値を設定します。上位のモードの境界値は、下位のモードの境界値よりも大きい必要があります。

状態	モード	説明
キュー滞留数がモード0の閾値以下	モード0	話者リストの話者のパラメタに従って発声を行う。
キュー滞留数がモード1の閾値を越えた	モード1	キュー内のメッセージ数が指定の数を超えると、話者リストの話者の話速パラメタを1.3倍に変更して発声を行う。
キュー滞留数がモード2の閾値を越えた	モード2	キュー内のメッセージ数が指定の数を超えると、話者リストの話者の話速パラメタを1.5倍に変更して発声を行う。
キュー滞留数がモード3の閾値を越えた	モード3	キュー内のメッセージ数が指定の数を超えると、話者リストの話者の話速パラメタを1.7倍に変更しテキストの長さを24文字に切捨てして発声を行う。
キュー滞留数がモード4の閾値を越えた	モード4	キュー内のメッセージ数が指定の数を超えると、話者リストの話者の話速パラメタを1.9倍に変更しテキストの長さを12文字に切捨てして発声を行う。
キュー滞留数がモード5の閾値を越えた	モード5	キュー内のメッセージ数が指定の数を超えると、キューのテキスト（メッセージ）をすべてクリアして新たにキューへの登録を開始する。

## 呟き定義編集

偽装ちゃん22v が何も発声せず指定の時間が経過すると呟くメッセージを指定できます。メインウインドウにあるボタン「呟き制御」を押すと表示されます。

適用	咳くメッセージ
<input checked="" type="checkbox"/>	いらっしゃいませー
<input checked="" type="checkbox"/>	コメントくださいねー
<input type="checkbox"/>	

発声タイミングを登録し、各タイミング毎に発声させるメッセージ一覧と発声で使用する話者リストを指定します。

話者、メッセージ、はそれぞれランダムに決定されます。

「咳き機能を利用する」のチェックを外すと、咳き機能は停止します。

## タイトル

ウィンドウタイトルが気に入らないという人のために用意されました。メインウィンドウにあるボタン「タイトル」を押すと表示されます。

## コメントジェネレータファイルパス

コメジェネで生成するcomment.xml互換のXMLファイルを生成するパスを指定します。メインウィンドウにあるボタン「コメGENファイル」を押すと保存先のフォルダとファイル名を指定するためのダイアログが表示されます。

## リスナ初期化

メインウィンドウにあるボタン「リスナ初期化」を押すとリスナ設定の内容が初期化されます。他の設定は初期化されません。

# 他アプリとの連携

## 自分のアプリから連携

IPC, Socket, HTTP による接続のサンプルは棒読みちゃんに同梱されていますのでそちらをご覧ください。

## MultiCommentViewer から使う

MultiCommentViewerの棒読みちゃん連携ではFakeChan22v.exeを選択できません。FakeChan22v.exeをコピーしてBouyomiChan.exeにリネームします。

## NiconamaCommentViewer から使う

NCVは自身で棒読みちゃん(偽装ちゃん)を起動しなかった場合、定期的に棒読みちゃんプロセス存在確認を行っているようです。

BouyomiChan.exeがプロセスに存在しないとコメント読み上げの度に棒読みちゃん(偽装ちゃん)を立ち上げようとしています。

FakeChan22v.exeをコピーしてBouyomiChan.exeにリネームし、NCVのオプションでこれを指定します。

## ツイキャスコメントビューアー（閲覧君）から使う

オプションでソケット通信を選択します。FakeChan22v.exeを起動すると読み上げが始まります。

ツイキャスコメントビューアーは棒読みちゃんの話者指定に声質1番(女性1)を指定するようなので、必要なら話者リストに2話者以上を登録してください。

[棒読みちゃん](#), [偽装ちゃん22v](#), [コメント読み上げ](#)

From:  
<https://wiki.hgotoh.jp/> - 努力したWiki

Permanent link:  
<https://wiki.hgotoh.jp/documents/tools/fakechan/fakechan22v/tools-209>

Last update: 2025/10/04 03:41

