

TuneEditor 4.07利用マニュアル

村上 雅章

September 5, 2022

Contents

1	はじめに	5
1.1	TuneEditor 4.xとは？	5
1.2	動作環境	5
1.3	ライセンス	6
2	使い方	7
2.1	行間表記の入力	7
2.1.1	はじめに	7
2.1.2	テキストの入力	7
2.1.3	他のアプリでコピーした文字（群）のペースト	8
2.1.4	国際音声記号（IPA）や全角文字などの特殊な文字の入力	8
2.1.5	ドットの大きさと高さの指定	9
2.1.6	音調核でのイントネーション変化を指定する	9
2.1.7	ドットを移動する	10
2.1.8	ドットを非表示にする、または表示状態に戻す	10
2.1.9	新たにドットを追加する	10
2.1.10	中央線を表示する、または非表示にする	11
2.1.11	区切り線を追加する	11
2.2	TSMの入力	13
2.2.1	各種記号の入力	13
2.2.2	下線の引き方	14
2.2.3	下線の消し方	15
2.2.4	その他の注意点	15
3	各種の設定	17
3.1	URLパラメーターからの変更	17
3.2	メンテナンスダイアログボックスからの変更	18
3.3	設定情報を書き込んだ.cfgファイルからの変更	18
3.4	指定できるパラメーター	18
3.4.1	描画エリアの全体レイアウト	18
3.4.2	TSM領域に関するパラメーター	19
3.4.3	行間表示領域に関するパラメーター	19
3.4.4	キーボード割り当て	20
3.5	各種パラメーター（まとめ）	21
4	アップデートで追加された機能	27
4.1	バージョン4.07で追加された機能	27

- 4.1.1 テキスト未入力域で行間表記の上／下（および点線）の
表示を抑止する 27
- 4.1.2 画面の左端から指定ピクセル数分だけ、上／下線（および
点線）の表示を抑止する 27

Chapter 1

はじめに

1.1 TuneEditor 4.xとは？

TuneEditorとは、英語の文字表現や発音記号（音声記号）だけでは表しきれないプロソディー（音の強弱や高低、長短）を視覚情報として表記するためのアプリケーションであり、イギリスの音声学研究におけるロンドン学派の表記法を簡単に入力できるようにすることを第1の目的にしています。

サポートしている表記法は、英語の文章中に付加記号（Tone Stress Mark：TSM）を記述する方式（以下、**TSM表記**と略）と、音の高低を表す上下線の間に黒いドット（音の高低を高さで、音の強弱を大きさで表し、音の長短／リズムを（ざっくりと）その間隔で表現する）を記述して表現する方式（以下、**行間表記**と略）です。

なお入力後のイメージはjpeg形式の画像ファイルとして、また作業中の内容をプロジェクトファイル（拡張子は「.ti4」）として保存できます。

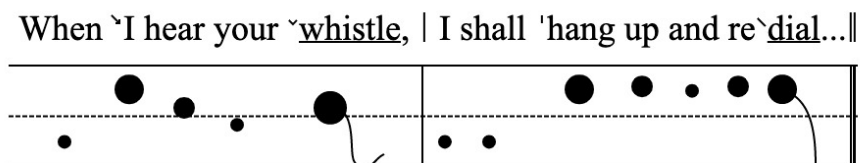


Figure 1.1: TuneEditorの出力サンプル

1.2 動作環境

このアプリケーションはブラウザアプリです。動作確認はmacOS上のGoogle Chromeで行っていますが、基本的にはモダンなブラウザであれば動作するはず

追記 : FireFoxではabout:configの画面で

```
dom.textMetrics.fontBoundingBox.enabled=true
```

と設定しておく必要があります。

また、ディスプレイ画面は横幅900ピクセル以上がおすすめですが、ブラウザの縮小表示機能を使えばもう少し小さいウィンドウ領域でも使用できるはずです。

1.3 ライセンス

本アプリケーションはMIT Licenseを採用したフリーソフトウェアであり、無償で利用することができます（利用の結果で発生する問題等については無保証ですが、希望を述べてもらうと対応するかもしれません ^^;）。

Chapter 2

使い方

2.1 行間表記の入力

2.1.1 はじめに

TuneEditorはウィンドウ上部に数多くのボタン類を配置しているため、編集領域が狭くなっています。このため使用に先立って、ブラウザの表示縮小コマンドを用いて表示サイズを縮小しておくのがおすすめです。

2.1.2 テキストの入力

テキストの入力は通常のエディターと同様であり、キーボードから入力した文字がそのままテキスト領域に表示されていきます。そして、その入力に合わせて行間表記のドットが自動的に追加されていきます。なお、ドットの追加規則は以下の通りです。

ドットの追加規則

- 入力した文字が**音節核候補文字**（“aeiouy”というアルファベット6文字（大文字および小文字）と、“`iɯɰɨɱʋøəøɤæœɜɛɛɔɛɔɔɔ`”という国際音声記号（IPA）24文字）である場合、直前および直後にある文字の**双方**が音節核候補文字**以外**の文字であれば、該当位置にドットを挿入します。
- 入力した文字が音節核候補文字**以外**である場合、直前および直後にある文字の**双方**が音節核候補文字であれば、該当位置の次の文字位置にドットを挿入します。

※ なお、行頭の直前、そして行末の直後には架空の**非音節核候補文字**があると想定して動作します。

2.1.3 他のアプリでコピーした文字（群）のペースト

他のアプリでコピーした文字（群）をペーストするには、以下の操作を実行します。

1. TuneEditorの画面右上にある水色の領域をクリックして、ピンク色に変える。
2. その状態で、command+V (macOSの場合)、またはCtrl+V (WindowsやLinuxの場合) を押下する。

なお、いったんピンク色に変えた状態で、ペースト処理を中止したい場合には、画面上の他の領域をクリックすれば、領域は水色に戻ります。

コピー&ペースト操作に関する舞台裏の話

TuneEditorのようなブラウザ上で動作するウェブアプリには、コピー&ペースト機能に厳しい制限が課されています。というのも、ウェブアプリは、ウェブサイトにアクセスするだけで自動的に実行されるようになっているため、コピー&ペーストに何らかの制限を課しておかなければ、ユーザーがデリケートな情報（パスワードや個人情報など）をクリップボードにコピーしている状態で、悪意のあるURLにアクセスした場合、あっという間にその情報が犯罪者たちのもとに送信されてしまうためです。

この制限ゆえに、ウェブアプリ内で実行するコピー&ペースト処理は通常のコピー&ペースト操作とは異なり、ユーザーの許可を必要とするようになっています。しかも標準化策定の遅れもあって、2022年6月時点ではすべてのブラウザで標準APIが実装されるに至っていません。

このため、TuneEditorではコピー&ペースト用の標準APIを用いるのではなく、ブラウザがあらかじめ用意している、画面作成用の部品を利用してウェブアプリ内にクリップボードの内容を送り込むという方式を採用しています。

早い話、ウィンドウ右上の水色／ピンク色の領域は、HTMLレベルで用意されている単なるテキスト領域（TEXTAREA）です（テキスト領域へのペーストは、従来のコピー&ペーストと同じ操作が可能になっているのです）。ただ、このテキスト領域はフォーカスが当たっていない時には背景色、文字色ともに水色に、フォーカスが当たっている時には背景色、文字色ともにピンク色になるよう設定しています（つまり単なる「インジケータ」兼データの入れ物として使っています）。そして、この領域に入力があった場合、TuneEditorは文字（群）が入力されたと判断し、その内容を取り込むようになっているのです。

2.1.4 国際音声記号（IPA）や全角文字などの特殊な文字の入力

TuneEditorはキーボードの押下を直接読み取って動作するため、キーボード上に刻印されている文字以外を入力することはできません。つまり、国際音声

記号 (IPA) やかな漢字変換の結果 (つまり漢字などの全角文字) を入力することはできません。

そういった文字を入力するには、他のアプリケーション上で作成した文字 (群) をコピー&ペーストによって入力することになります。操作の詳細は8ページを参照してください。

2.1.5 ドットの大きさと高さの指定

ドットの大きさと高さを指定するには、ウィンドウ上部のStressという見出しの下にある3x9列のボタン群を使用します。操作方法は以下の通りです。

1. 指定したいドットをクリックすると、そのドットの表示色が赤くなり、フォーカスが当たる¹とともに、テキストカーソルも対応付けられている音節核候補文字の先頭に移動します。
2. 必要なドットサイズ (大/中/小のいずれか) と高さに該当するボタンを押します。

なお、ボタンを押下すると、次のドットにフォーカスが移動します。この移動を抑止したい場合には、Shiftキーを押しながらボタンを押してください。

また、直前のドットにフォーカスを移動させたい場合にはcommand+Z (macOS使用時)、またはCtrl+Z (WindowsまたはLinux使用時) を押下するか、ウィンドウ画面上部の [◀] をクリックしてください。

さらに、直後のドットにフォーカスを移動させたい場合にはcommand+shift+Z (macOS使用時)、またはCtrl+Shift+Z (WindowsまたはLinux使用時) を押下するか、ウィンドウ画面上部の [▶] をクリックしてください。



2.1.6 音調核でのイントネーション変化を指定する

ドットには、音調核の単音節内でのピッチ移動を表現するために5種類²の「尻尾」を付加することができます。これにはToneという見出しの下にある6つのボタンを使用します。

Low/High-Fall : 下降ピッチ

Rise-Fall : 上昇後下降ピッチ

Low/High-Rise : 上昇ピッチ

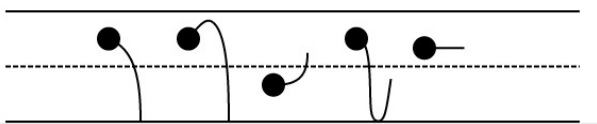
Fall-Rise : 下降後上昇ピッチ

¹もう一度クリックするとフォーカスは外れます。

²TSMでは7種類の記号が用意されていますが、Low-FallとHigh-Fall、そしてLow-RiseとHigh-Riseはドット自体のピッチの高さで区別できるため、5種類となっています。

Mid-Level : 平坦ピッチ

-cancel pattern- : ピッチ無し (リセット)



なお、ドットの大きさと高さを変更するボタンを押した場合、ドットのフォーカスは自動的に次のドットに移り、そのドットが赤色で表示されますが、この操作の直後にピッチ移動ボタンを押した時に限り、ピッチ移動の適用対象は今高さと大きさを指定したドットになります。それ以外の場合は、まず適用対象のドットをマウスでクリックした上で、ピッチ移動ボタンを押してください。

2.1.7 ドットを移動する

ドットの移動はマウスのドラッグで可能です。

2.1.8 ドットを非表示にする、または表示状態に戻す

TuneEditorでは入力した文字のパターンに従って、ドットを追加するようになっています。このため、例えば“cake”といった1音節の単語を入力した場合であっても、2文字目の“a”に対応した適切なドットと、4文字目の“e”に対応した不要なドットが表示されてしまいます。こういった不要なドットを非表示にするには、以下の手順を実行します。

1. 非表示にしたいドットをクリックし、赤色で表示させます。
2. [Make item invisible] ボタンを押します。

また、いったん非表示にしたドットは、以下の手順で表示状態に戻すことができます。

1. [Make item visible again] ボタンを押すと、非表示にしたドット（群）が青色で表示されます。
2. 表示状態に戻したいドットをクリックします。

2.1.9 新たにドットを追加する

TuneEditorでは入力した文字の種類と組み合わせに従って、ドットを追加するようになっています。このため、例えば“10”と入力した場合にはどこにも音節候補文字がないため、ドットが表示されません。こういったケースに対応するための、以下のような姑息な (!) 手段が用意されています。

1. ドットを追加したい場所の近傍にある、音節候補文字ではない文字が2つ連続している場所を見つけ出します。
2. それら文字の間でマウスをクリックし、テキストカーソルを移

動した後、[esc] キーを押してコマンドメニューを表示させ、「aa」（Add A Toneコマンド）と入力します。

音節核候補文字のない場所でのドット表示

TuneEditorでは音節核候補文字のない場所でドットを表示させるため、「テキスト上は表示されない、すなわち**横幅がゼロ**の文字を用意して、この文字を音節核候補文字として扱うようにしています。そして、[esc]-「aa」コマンドでその文字を入力するようにしています。

つまり、音節核候補文字ではない文字の並びの間に「表示されない音節核候補文字」を挿入することで、ドットが追加されるように仕向けているのです。

このやり方の弊害として、テキストカーソルを矢印キーで1文字ずつ移動させると、表示されない文字のある部分で、カーソルが足踏みするように見えるという点があります（許してください）。

2.1.10 中央線を表示する、または非表示にする

行間表記の中間にデフォルトで描画される破線は、ウィンドウ上部の [Centreline] チェックボックスを用いて表示／非表示を切り替えることができます。

この中央線を描画するというアイデアは、長瀬慶来先生より久保岳夫先生経由で頂いたコメントに基づいております。この場を借りて感謝いたします。

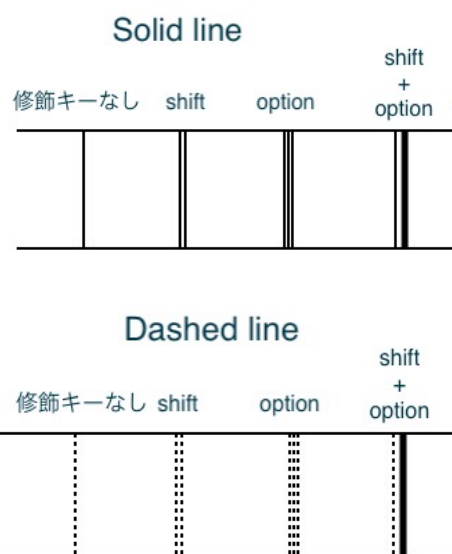
2.1.11 区切り線を追加する

行間表記領域には8種類の区切り線を追加することができます。これらはマウスのドラッグで移動でき、ドットと同様の操作で非表示にしたり、再表示させることができます。

行間表記の区切り線は8種類をサポートしており、以下のようにして入力します。

1. ウィンドウ上部の [Separator (Solid line)] (実線表示) または [Separator (Dashed line)] (破線表示) をマウスでクリックする。
2. 区切り線を入力したいところでマウスをクリックする。

なお、マウスをクリックする際に「shift」キー（二重線）や、「option」キー（三重線）、「shift+option」キー（変則二重線）を押すことで区切り線の形状を変えることができます。



注意点：なおTuneEditor Ver. 4.01では、区切り線の位置管理を文字種別とは独立したかたちで行うようになったため、テキスト内容を変更しても、その変更に従って区切り線の描画場所がシフトしていきようにはなっていません（気が向けば対応します）。

2.2 TSMの入力

TSMはテキスト領域に付加していく記号です。行間表記は不要でTSM表記のみ必要だという場合、ウィンドウ上部にある [Use Graphics] チェックボックスのチェックを外してください。

2.2.1 各種記号の入力

TuneEditorでは以下のTSM記号をサポートしており、すべて「esc」キーを押下した際に表示されるコマンドメニューから行えるようになっています。

TSM Symbols: `— ‘ ° \ , . / \ / ^ v > | ||`

また、公式に定義されていないLow-Prehead記号 (“_”) や、複数の区切り線のバリエーションも使用できます。

Additional Symbols: `| | | | || || || ||| ||| | | | |`

なお、「esc」キーで表示されるメニューは、もう一度「esc」キーを押すか、2文字を入力すると閉じ、その2文字がコマンドとして定義されている場合に、該当コマンドを実行します。

以下は、公式のTSM記号を入力する際に用いるコマンドの一覧です。

記号	コマンド	意味
—	HP	高前頭部
’	H1	高い音の強勢
°	H2	高い音の準強勢
\	H3	下降調
,	L1	低い音の強勢
.	L2	低い音の準強勢
/	L3	上昇調
\	HF	高下降調 (音調核)
/	HR	高上昇調 (音調核)
\	LF	低下降調 (音調核)
/	LR	低上昇調 (音調核)
^	RF	上昇下降調 (音調核)
v	FR	下降上昇調 (音調核)
>	ML	平坦調 (音調核)
	IP	イントネーション句の区切り
	FS	イントネーション句の区切り (文末等)

非公式な記号を入力するコマンドは、メニューに表示されていませんが、次のようになっています。

lp 低前頭部の記号です（これは高前頭部との違いを際立たせたい特殊な場合にしようできます）

i0, i1, i2, i3 区切り線の上に赤で「0~3」の数字を追記した記号です

s1, s2, s3, s4 実線の区切り線で、行間表記の区切り線の4パターンに対応した記号です（12ページを参照）。

d1, d2, d3, d4 破線の区切り線で、行間表記の区切り線の4パターンに対応した記号です（12ページを参照）。

コマンド	意味
LP	低前頭部記号（高前頭部との違いを際立たせたい場合に使用）
i0	実線の区切り線の上に赤で「0」を表示したもの
i1	実線の区切り線の上に赤で「1」を表示したもの
i2	実線の区切り線の上に赤で「2」を表示したもの
i3	実線の区切り線の上に赤で「3」を表示したもの
s1	実線の区切り線（「IP」と同じ）
s2	実線の区切り線（「FS」と同じ）
s3	実線の区切り線（3重線）
s4	実線の区切り線（変則2重線）
d1	破線の区切り線（1本線）
d2	破線の区切り線（2重線）
d3	破線の区切り線（3重線）
d4	破線の区切り線（変則2重線）

2.2.2 下線の引き方

以下の手順で下線を引くことができます。

- 以下のいずれかの方法で、下線を引きたい部分を選択する。
 - 選択範囲の一方の端にテキストカーソルを置き、「Shift」キーを押しながら、矢印キーを押して選択範囲を広げていく。
 - テキスト領域をマウスでドラッグする。
 - 選択範囲の一方の端にテキストカーソルを置き、「Shift」キーを押しながら、もう一方の端をマウスでクリックする。
- 「esc」キーを押した後、「su」（Set Underline）と入力する。

2.2.3 下線の消し方

いったん引いた下線は、下線を引いた際と同じ方法で下線を消去したい領域を選択し、「esc」キーを押した後、「cu」（Clear Underline）と入力することで消去できます。

2.2.4 その他の注意点

テキスト行のコピー&ペーストは、ブラウザー上のTuneEditorインスタンスが実行されているタブ内でしか実行できません。またその際には、TSM記号をコピーすることはできません（記号部分は空白として処理されます）。

Chapter 3

各種の設定

TuneEditorは出力イメージ (jpegファイル) のレイアウトや文字／ドットの設定、キーボード割り当てなどを変更できるようになっています。

設定するための方法は、以下の3種類を用意しています。

URLパラメーターからの変更： アプリ起動時のURLで設定を指定する方法です。URLパラメーターを指定した状態でブックマークとして登録しておけば、常にその設定で起動／使用できるようになります。

メンテナンスダイアログボックスからの変更： アプリケーション実行時にメンテナンス用のダイアログボックスから変更したい設定をひとつずつ指定する方法です。この設定はブラウザのタブをクローズ／再ロードするまで有効となります。主な用途としては、実験的にパラメーターを変更して、その効果を確認するというものであり、次に説明する.cfgファイルを作成する時に利用することができます。

設定情報を書き込んだ.cfgファイルからの変更： ファイル入力ボタンから.cfg形式のファイルを読み込み、複数のパラメーターを一度に指定する方法です。この設定はブラウザのタブをクローズ／再ロードするまで有効となります。

設定できるパラメーターについては、18ページを参照してください。

注意点： パラメーターのキーワードは大／小文字を意識しています。間違えると機能しないのでご注意ください。

3.1 URLパラメーターからの変更

TuneEditorのURLの後ろには、「**キーワード=値**」という形式でパラメーターを記述していくことができるようになっています。この際、URLとパラメーター

ターの間は「?」で区切り、複数のパラメーターを続けて指定するにはそれぞれのパラメーター間を「&」で区切ってください。また、値に空白(“ ”)が含まれる場合には、空白を「%20」に変換してください。

例

```
https://～省略～/tuneEditor4.html?DotSSize=6&DotMSize=8&DotLSize=12
```

※ この例は最小サイズのドット径を6、中サイズのドット径を8、最大サイズのドット径を12にするというものです。

3.2 メンテナンスダイアログボックスからの変更

実行中に「Ctrl+K」キーを押下することで、Enter override parameter (key=value format)と表示されたメンテナンス用のダイアログボックスがオープンします。ここから指定したいパラメーターのキーワードと値を「=」で区切って入力します。

3.3 設定情報を書き込んだ.cfgファイルからの変更

テキストエディターを使用して、設定情報ファイルを作成します。設定ファイルの形式は、「**キーワード=値**」という指定を1行ずつ入力します。

例

```
DotSSize=6  
DotMSize=8  
DotLSize=12
```

3.4 指定できるパラメーター

3.4.1 描画エリアの全体レイアウト

TuneEditorは出力イメージを作成するにあたり、最小の幅 (CanvasMinWidth) と最小の高さ (CanvasMinHeight) という設定値から**最小限の描画領域**を作成し、その上下左右に余白 (CanvasTopMargin, CanvasBottomMargin, CanvasLeftMargin, CanvasRightMargin) を確保するようになっています。

なお、この最小限の描画領域の幅は、入力された行が長くなっていくと必要に応じて自動拡張されます。また、高さも改行が入力されるたびに、TSM行の高さ (TSMHeight) と、行間表記を使う場合にはその高さ (DotsAreaHeight) が加味されて自動的に拡張していくようになっています。

全体レイアウトに関するパラメーター群とデフォルト値は21ページにまとめてあります。

3.4.2 TSM領域に関するパラメーター

TSM領域を管理するパラメーターには以下のものがあります。

TSMHeight TSM領域の高さをピクセル単位で指定します。

TSMBaseLineOffset 該当行のTSM領域の最上部から見た文字のベースラインの位置をピクセル単位で指定します。

FontName フォント名称を指定します。

FontSize フォントサイズを指定します。なお、この使用は非推奨です（このパラメーターは通常の文字のサイズを指定するものであるため、TSM記号のサイズは変更されませんつまり使用すると文字とTSM記号のバランスが崩れてしまうのです）。

TSM領域に関するパラメーター群とデフォルト値は21ページにまとめてあります。

3.4.3 行間表示領域に関するパラメーター

行間表示領域を管理するパラメーターには以下のものがあります。

DotsSwitch 行間表示の使用有無を「0」か「1」で指定します。

DotsAreaHeight 行間表示領域の高さをピクセル単位で指定します。

TopLineOffset 該当行の行間表示領域の最上部から見た上限ラインの位置をピクセル単位で指定します。

BottomLineOffset 該当行の行間表示領域の最上部から見た上限ラインの位置をピクセル単位で指定します。

DotSSize 最小サイズのドット径をピクセル単位で指定します。

DotMSize 中サイズのドット径をピクセル単位で指定します。

DotLSize 最大サイズのドット径をピクセル単位で指定します。

GlideMagFactor 現在未使用。

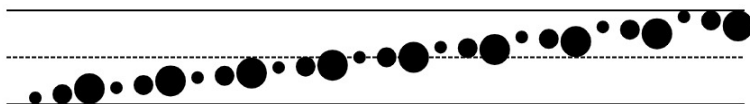
DotDistributionPattern ドットの分布パターンを「0」～「2」で指定します（後述）。

CentrelineSwitch センターラインの描画有無を「0」か「1」で指定します。

ドットの配置パターンについて

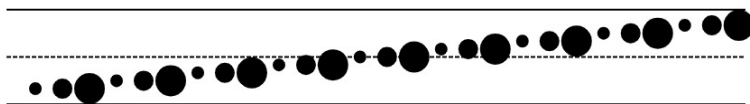
上下線の中にサイズの異なるドットを縦方向に配分しようとした場合、いくつかのパターンが考えられます。

- **設定「0」**：ドットの外周が上下線に接する位置を大／中／小のドットごとに計算し、中間の音は均等割りで配置する。



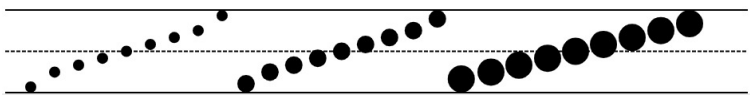
デメリット：この場合、同じ高さの音でもドットのサイズが異なっている場合、中心の高さが異なって見える。

- **設定「1」**：最大のドットの外周が上下線に接する際のドットの中心位置を算出し、すべてのサイズのドットでその中心位置を使用するようにする。なお中間の音は均等割りで配置する。



デメリット：この場合、大サイズ以外のドットの外周は上下線に接することがなくなる。

- **設定「2」**：ドットの高さが最低／最高の場合にドットの外周が上下線に接するようにし、それ以外の場合には設定「1」を採用する。



デメリット：この場合、上昇調／下降調でドットを並べた場合、大サイズ以外のドットが直線上に並ばないケースが出てくる。

なお、TuneEditorではこれら3つの戦略を起動時に選択することが可能になっています。

行間表示領域に関するパラメーター群とデフォルト値は22ページにまとめてあります。

3.4.4 キーボード割り当て

各種のキーボード割り当てでも変更することができますが、OSが規定しているキーボードショートカットと矛盾しないようなかたちで指定する必要があるため、思いがけない副作用に頭を抱えることになるかもしれません。

一般的な機能に対応付けられているキーボード割り当てパラメーター群とデフォルト値は24ページにまとめてあります。

また、ドット操作に関するキーボード割り当てパラメーター群とデフォルト値は25ページにまとめてあります。

3.5 各種パラメーター (まとめ)

Table 3.1: 全体レイアウトに関するパラメーター

キーワード	意味	デフォルト値
CanvasMinWidth	最小限の描画領域の幅 (ピクセル単位)	1000
CanvasMinHeight	最小限の描画領域の高さ (ピクセル単位)	240
CanvasTopMargin	全体画像の上部余白 (ピクセル単位)	30
CanvasBottomMargin	全体画像の下部余白 (ピクセル単位)	30
CanvasLeftMargin	全体画像の左余白 (ピクセル単位)	30
CanvasRightMargin	全体画像の右余白 (ピクセル単位)	30

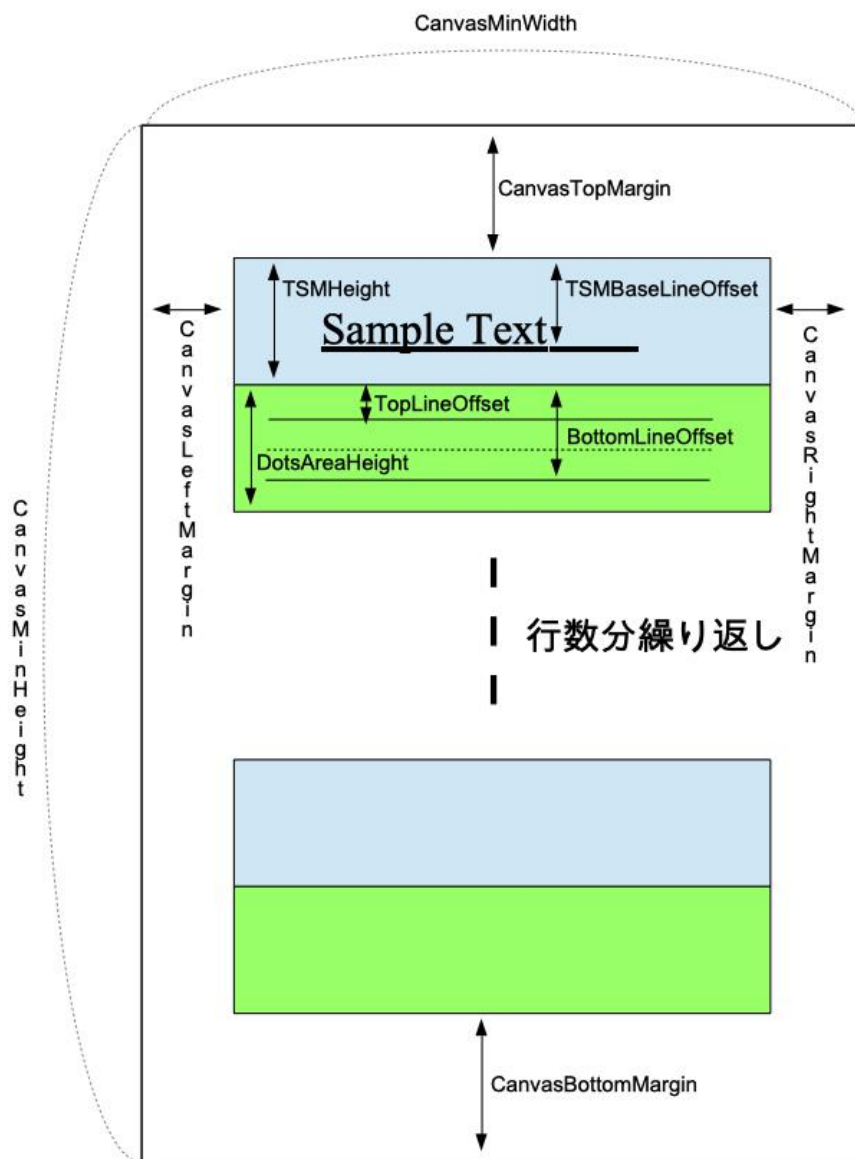
Table 3.2: TSM領域に関するパラメーター

キーワード	意味	デフォルト値
TSMHeight	TSM領域の高さ (ピクセル単位)	55
TSMBaseLineOffset	文字のベースライン位置 (ピクセル単位) (TSM領域の最上部からのオフセット)	40
FontName	フォント名	Times New Roman
FontSize	フォントサイズ (使用は非推奨)	27

Table 3.3: 行間表示領域に関するパラメーター

キーワード	意味	デフォルト値
DotsSwitch	行間表示の有無（「0」は不要、「1」は要）	1
DotsAreaHeight	行間表示の高さ （ピクセル単位）	130
TopLineOffset	行間表示開始位置と上限線のオフセット （ピクセル単位）	10
BottomLineOffset	行間表示開始位置と下限線のオフセット （ピクセル単位）	98
DotSSize	最小サイズのドット径（ピクセル単位）	5
DotMSize	中サイズのドット径（ピクセル単位）	8
DotLSize	最大サイズのドット径（ピクセル単位）	12
GlideMagFactor	尻尾の倍率係数（実数）-現在未使用	1.00
DotDistributionPattern	ドット分布パターン （「0」～「2」:20ページを参照）	2
CentrelineSwitch	センターラインの有無 （「0」は不要、「1」は要）	1

Table 3.4: キャンバス関連のパラメーターの関連図



キャンバス内の描画位置を決める手順は以下の通り

1. 最小限のサイズとしてCanvasMinWidth×CanvasMinHeightの領域を用意する(必要に応じて拡張される)。
2. 上下左右のマージン(CanvasTopMargin、CanvasBottomMargin、CanvasLeftMargin、CanvasRightMargin)を確保する。
3. テキストの各行ごとにTSMHeight(と行間表示を使用する場合はDotsAreaHeight)を確保する。
4. テキストのベースラインは、各行のテキスト領域上部がTSMBaseLineOffset分下がった位置。
5. 行間表記の上限のラインは、各行のSM領域上部からTopLineOffset分下がった位置に、下限のラインは同じくBottomLineOffset分下がった位置になる。

Table 3.5: 一般機能に関するキーボード割り当てパラメーター

キーワード	意味	デフォルト値
CursorLeft	テキストカーソルを1文字分左に	ArrowLeft
CursorRight	テキストカーソルを1文字分右に	ArrowRight
AnchorAndCursorLeft	テキスト選択範囲を左に1文字分拡張	Shift+ArrowLeft
AnchorAndCursorRight	テキスト選択範囲を右に1文字分拡張	Shift+ArrowRight
CursorUp	テキストカーソルを1行上に	ArrowUp
CursorDown	テキストカーソルを1行下に	ArrowDown
AnchorAndCursorUp	テキスト選択範囲の1行上に拡張	Shift+ArrowUp
AnchorAndCursorDown	テキスト選択範囲の1行下に拡張	Shift+ArrowDown
CursorHead	テキストカーソルを行頭に	Ctrl+q
CursorTail	テキストカーソルを行末に	Shift+Ctrl+q
CursorTail2	テキストカーソルを行末に	Ctrl+w
CursorTop	テキストカーソルを文書頭に	Meta+ArrowUp
CursorBottom	テキストカーソルを文書末に	Meta+ArrowDown
CursorPrevWord	テキスト選択範囲を1ワード分前に	Shift+Ctrl+ArrowLeft
CursorNextWord	テキスト選択範囲を1ワード分後に	Shift+Ctrl+ArrowRight
Backspace	テキストカーソルの直前1文字を削除	Backspace
DeleteChar	テキストカーソルの直後1文字を削除	Delete
CopyRegion	選択範囲をアプリ内の バッファにコピー	Meta+c
CopyRegion2	選択範囲をアプリ内の バッファにコピー	Ctrl+c
CutRegion	選択範囲をアプリ内の バッファにコピーしカット	Meta+x
CutRegion2	選択範囲をアプリ内の バッファにコピーしカット	Ctrl+x
Paste	アプリ内のバッファからペースト	Meta+v
Paste2	アプリ内のバッファからペースト	Ctrl+v
SelectAll	テキスト全体を選択	Meta+a
SelectAll2	テキスト全体を選択	Ctrl+a
MaintenanceHatch	パラメーター設定ダイアログボックスの オープン	Ctrl+k

Table 3.6: ドット操作に関するキーボード割り当てパラメーター

キーワード	意味	デフォルト値
StopAutoMove	ドットの自動移動を抑止するキー	Shift
WiderToneTail	ドットの尻尾の幅を拡大する	Shift+Meta+ArrowRight
WiderToneTail2	ドットの尻尾の幅を拡大する	Ctrl+ArrowRight
NarrowerToneTail	ドットの尻尾の幅を縮小する	Shift+Meta+ArrowLeft
NarrowerToneTail2	ドットの尻尾の幅を縮小する	Ctrl+ArrowLeft
RaiseToneTail	ドットの尻尾の高さを拡大する	Shift+Meta+ArrowUp
RaiseToneTail2	ドットの尻尾の高さを拡大する	Ctrl+ArrowUp
LowerToneTail	ドットの尻尾の高さを縮小する	Shift+Meta+ArrowDown
LowerToneTail2	ドットの尻尾の高さを縮小する	Ctrl+ArrowDown
RaiseToneTailEnd	ドットの尻尾末尾の高さを上げる	Shift+Ctrl+Meta+ArrowUp
LowerToneTailEnd	ドットの尻尾末尾の高さを下げる	Shift+Ctrl+Meta+ArrowDown
NextTone	次のドットにフォーカスを移す	Shift+Meta+z
NextTone2	次のドットにフォーカスを移す	Shift+Ctrl+Z
PrevTone	前のドットにフォーカスを移す	Meta+z
PrevTone2	前のドットにフォーカスを移す	Shift+Ctrl+Z

Chapter 4

アップデートで追加された機能

4.1 バージョン4.07で追加された機能

4.1.1 テキスト未入力域で行間表記の平行線表示を抑止する

「FullBaseline=0」と指定すると、テキスト未入力域では行間表記の平行線が表示されないようになります。指定しない、あるいは「0」以外を指定すると、従来どおり画像ファイルの右端まで平行線が描画されます。

4.1.2 画面の左端から指定ピクセル数分だけ、平行線の表示を抑止する

複数人の対話ダイアログを表記するような場合、画面の左端に話者を識別する名称を表記する場合があります。この場合、そういった名称には行間表記を使用しないため、平行線は不要となるはずですが、

こういった用途のために「sf」パラメーターを追加しました。このパラメーター（Speaker Fieldの略です）を指定することにより、画面の左端から指定したピクセル数分だけ、平行線の描画が抑止されます。

なお、表示が抑止されるのは平行線だけであるため、話者の識別名称に音節核候補文字が含まれている場合、対応するドット（群）が表示されます。この場合、[Make Item Invisible] コマンドを使用して、ドットを消し込んでください。

使用に際しては、話者名込みのかたちでダイアログを入力した後、Control+kを押下し、「sf=50」などと入力してみて、話者識別名の下が表示がどの程度抑止されるのかを確認し、適当な数値を探し出してください。