

RasterFontGen

# 概要

RasterFontGen は、「ラスタライズされた」TrueType フォントを作成するためのツールです。RasterFontGen を使用して、後でゲームに読み込むことができる .rasterfont ファイルを作成し、RasterFont API を使用してレンダリングできます。.rasterfont 形式は特に Scorpio Devkit フォント パネルを念頭に置いて開発されました。特に、.rasterfont は CPU でテキストをレンダリングするのに適した軽量のテキスト レンダリング システムをサポートしています。

# RasterFontGen のビルド

RasterFontGen はソース コードとして提供されています。このツールをビルドするには、単純に RasterFontGen.sln を開いて Visual Studio からビルドします。

# RasterFontGen の使用

完全なコマンド ライン オプションを取得するには、以下を使用します。

>RasterFontGen.exe /?

RasterFontGen は以下の機能をサポートします。

* シンボル フォントを含めて、あらゆる TrueType フォントで動作します。ツールを実行する PC にフォントをインストールする必要があります。
* さまざまなフォントの高さをピクセル単位で指定できます。
* Unicode コードの範囲を柔軟に指定できるように、複数のコード範囲がサポートされています。

高さオプションは、高さを「論理単位」(ピクセルと考える) で指定するために使用され、「プリンタのポイント」とは関係ありません。つまり、-h 16 を指定すると、高さ 16 ピクセルのフォントになります。これは、フォントでサポートされているすべてのグリフに対して考慮される、最も高いアセンダーから最も深いディセンダーまでの距離が 16 ピクセルになることを意味します。

書体に値を指定しないと、このツールではシステム フォント選択アルゴリズムを使用して、他の指定された引数に使用できるフォントを見つけます。書体名のスペルを間違えた場合、ツールはシステム フォント選択アルゴリズムを使用して、他の指定されたパラメーターに適したフォントを見つけます。必要な書体を取得できない場合は、その書体を指定するために必要な文字列を正確に指定していない可能性があります。以下は、書体名を指定するのに役立つ推奨事項です。

* どのフォントをインストールしたか確認するために、コントロール パネルの Fonts フォルダに移動します。これをガイドとして使用して、フォントの正しい名前 (つまり、-tf 引数に指定する値) を選択してください。
* Fonts フォルダには、単一のフォントとフォント ファミリを表す別々のアイコンがあります。探している特定の書体の「校正ページ」が開かれるまでドリルダウンしてください。校正ページに「Windows でコンピューターの世界が広がります。0123456789」というテキストが表示されます。このページには、使用する必要のある書体名が表示されます。(例： 「Algerian Regular」ではなく「Algerian」を使用してください。)

フォントのコード範囲を指定しないと、RasterFontGen は指定されたフォントでサポートされているすべてのグリフを生成します。通常は、結果の .rasterfont ファイルに生成されるグリフの数を制限するために、1 つ以上の -cr： 仕様を使用します。-cs フラグはシステム フォント選択アルゴリズムへのヒントとして使用され、生成されたグリフのセットのコード範囲を制限するために使用されるべきではないことに注意してください。

サポートされていないコード ポイントのグリフを生成しようとすると、警告が表示されます。特定のコード ポイントをサポートするフォントを探している場合は、このような警告に注意してください。それ以外の場合は、このような警告を無視しても大丈夫です。

# 例：

次の例では、Lucida Console から 16 論理単位 (ピクセル) の高さで .rasterfont ファイルを作成します。フォントでサポートされているすべてのグリフも含まれます (多数存在します)。

>RasterFontGen -tf “Lucida Console” -h 16 –of LC.rasterfont

次の例では、シンボル フォントから特定のコード範囲 (-cr を使用して指定) を使用して .rasterfont ファイルを作成します。

>RasterFontGen –tf “Segoe Xbox MDL 2 Assets” –h 16 –of Symbols16.rasterfont -cr:0xE3AF-0xE3B2 -cr:0xE48B-0xE48C

次の例では、ANSI のコード ポイントの範囲 (おそらく、より多数のグリフ) を対象とする Algerian Regular 書体を使用して .rasterfont ファイルを作成します。

>RasterFontGen -tf "Algerian" -cr:0x00-0xFF -h 16 -of AR.rasterfont

# 実装上の注意

このツールのコマンド ラインには、.rasterfont 形式でサポートされているよりも多数のオプションがあります。これは、コマンドラインが LOGFONT 構造のすべてのフィールドに対応するオプションを公開しているためです。この設計は、多少の混乱を招くものの、最大限の柔軟性を得るために選択されました。おそらく、より難解なオプションを使用することはありません。

.rasterfont 形式では現在 1 ピクセルあたり 1 ビット形式のみがサポートされます。コマンド ライン オプションでは、品質オプション (-q) とピクセル深度オプション (-d) の両方がサポートされます。これらのオプションは、各グリフが .rasterfont 形式に「フラット化」される前に (GetGlyphOutline を介して) レンダリングされる方法だけに影響します。そのため、これらのフラグを使用しても、影響や改善が見られない可能性があります。

免責事項：.rasterfont 形式に基づく超高品質のテキスト レンダリング システムが作成されることは期待しないでください。.rasterfont 形式は CPU 上での軽量テキスト レンダリングのみを目的としており、特に Scorpio Devkit フロント パネル ディスプレイでの使用を対象としています。想定された範囲内で、結果に非常に満足していただけることでしょう。