Autodesk®
Scaleform®

Font and Text Configuration Overview

Scaleform 4.3
( ) Scaleform C++ API

: Maxim Shemanarev, Michael Antonov
: 2.2
: 2012 6 21

Autodesk®
GAMEWARE

Except as otherwise permitted by Autodesk, Inc., this publication, or parts thereof, may not be reproduced in any form, by any method, for any purpose.

Certain materials included in this publication are reprinted with the permission of the copyright holder.

The following are registered trademarks or trademarks of Autodesk, Inc., and/or its subsidiaries and/or affiliates in the USA and other countries: 123D, 3ds Max, Algor, Alias, AliasStudio, ATC, AutoCAD, AutoCAD Learning Assistance, AutoCAD LT, AutoCAD Simulator, AutoCAD SQL Extension, AutoCAD SQL Interface, Autodesk, Autodesk 123D, Autodesk Homestyler, Autodesk Intent, Autodesk Inventor, Autodesk MapGuide, Autodesk Streamline, AutoLISP, AutoSketch, AutoSnap, AutoTrack, Backburner, Backdraft, Beast, Beast (design/logo), BIM 360, Built with ObjectARX (design/logo), Burn, Buzzsaw, CADmep, CAiCE, CAMduct, CFdesign, Civil 3D, Cleaner, Cleaner Central, ClearScale, Colour Warper, Combustion, Communication Specification, Constructware, Content Explorer, Creative Bridge, Dancing Baby (image), DesignCenter, Design Doctor, Designer's Toolkit, DesignKids, DesignProf, Design Server, DesignStudio, Design Web Format, Discreet, DWF, DWG, DWG (design/logo), DWG Extreme, DWG TrueConvert, DWG TrueView, DWGX, DXF, Ecotect, ESTmep, Evolver, Exposure, Extending the Design Team, FABmep, Face Robot, FBX, Fempro, Fire, Flame, Flare, Flint, FMDesktop, ForceEffect, Freewheel, GDX Driver, Glue, Green Building Studio, Heads-up Design, Heidi, Homestyler, HumanIK, i-drop, ImageModeler, iMOUT, Incinerator, Inferno, Instructables, Instructables (stylized robot design/logo), Inventor, Inventor LT, Kynapse, Kynogon, LandXplorer, Lustre, Map It, Build It, Use It, MatchMover, Maya, Mechanical Desktop, MIMI, Moldflow, Moldflow Plastics Advisers, Moldflow Plastics Insight, Moondust, MotionBuilder, Movimento, MPA, MPA (design/logo), MPI (design/logo), MPX, MPX (design/logo), Mudbox, Multi-Master Editing, Navisworks, ObjectARX, ObjectDBX, Opticore, Pipeplus, Pixlr, Pixlr-o-matic, PolarSnap, Powered with Autodesk Technology, Productstream, ProMaterials, RasterDWG, RealDWG, Real-time Roto, Recognize, Render Queue, Retimer, Reveal, Revit, Revit LT, RiverCAD, Robot, Scaleform, Scaleform GFx, Showcase, Show Me, ShowMotion, SketchBook, Smoke, Softimage, Socialcam, Sparks, SteeringWheels, Stitcher, Stone, StormNET, TinkerBox, ToolClip, Topobase, Toxik, TrustedDWG, T-Splines, U-Vis, ViewCube, Visual, Visual LISP, Vtour, WaterNetworks, Wire, Wiretap, WiretapCentral, XSI.

All other brand names, product names or trademarks belong to their respective holders.

THIS PUBLICATION AND THE INFORMATION CONTAINED HEREIN IS MADE AVAILABLE BY AUTODESK, INC. “AS IS.” AUTODESK, INC. DISCLAIMS ALL WARRANTIES, EITHER EXPRESS OR IMPLIED, INCLUDING BUT NOT LIMITED TO ANY IMPLIED WARRANTIES OF

MERCHANTABILITY OR FITNESS FOR A PARTICULAR PURPOSE REGARDING THESE MATERIALS.

Autodesk Scaleform	:
	(Font and Text Configuration Overview) Autodesk Scaleform Corporation 6305 Ivy Lane, Suite 310 Greenbelt, MD 20770, USA www.scaleform.com info@scaleform.com (301) 446-3200
FAX	(301) 446-3199

はじめに　　のフォント 1
1.1　Flash 2
1.1.1 2
1.1.2 3
1.1.3 4
1.1.4 5
1.2　UI 6
パート　ゲームフォントライブラリの作成 8
2.1 8
2.1.1 9
2.1.2 11
2.2　gfxfontlib.swf 11
パート　多国語化のアプローチの選択 13
3.1 13
3.1.1 15
3.1.2 Scaleform Player 17
3.1.3 18
3.1.4 19
3.2 21
パート　フォントソースの設定 23
4.1 23
4.2　GFx::FontMap 25
4.3　GFx::FontLib 26
4.4　GFx::FontProviderWin32 26
4.4.1 27
4.4.2 28
4.5　GFx::FontProviderFT2 29
4.5.1 FreeType 30
4.5.2 30
パート　フォント レンダリングの設定 33
5.1 34

5.2 35
5.3 gfxexport 38
5.4 - gfxexport 38
5.5 40
5.6 42
パート　テキストのフィルタエフェクトと　拡張 44
6.1 44
6.2 46
6.3 47
6.4 ActionScript 47

はじめに　　のフォント

Scaleform　　HTML

GFX/SWF
FreeType 2

/　　/

Scaleform　　Flash® Studio
SWF

Flash Studio
Flash

Flash

Scaleform
GFx::Translator
GFx::FontLib

FreeType 2

Scaleform　　Flash Studio　Scaleform
Flash

Scaleform

Flash　Scaleform　　UI

Flash
Flash　　Flash
(Text Field)

UI
Flash　　UI

1.1 Flash のテキストとフォントの基礎

Flash Studio

UI

Flash Studio

Flash

GF 1 –

テキストフィールド

Flash

Flash Studio [ ]

ActionScript

GFx::Translator

2 ( )
a) b)

Arial [ ] [ ]

SWF Flash 8

( )
( )
( )

1
SWF

Flash Player

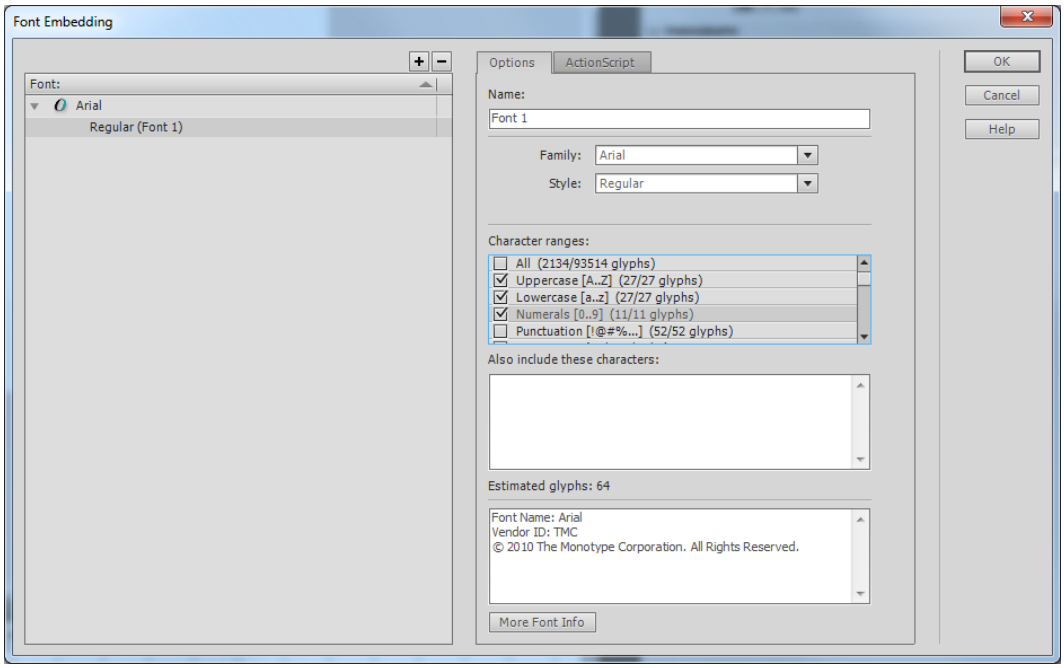
Scaleform

( ) ( )

[…]

文字の埋め込み

1 TextField
textfield
(Scaleform GFx::FontLib
) [...]
[ ]



JIS

Style Arial Arial Bold

Flash Studio
Using Fonts ( )

SWF/GFX
SWF/GFX

Scaleform (Xbox 360
PS3 Wii) PC (Windows Mac Linux)

( )

埋め込みフォントによるメモリの使用

Scaleform

	SWF	Scaleform **ランタイム**

1	1 KB	450 K
114 – +	12 KB	480 K
596 –	70 KB	630 K
7,583 –	2,089 KB	3,500 K
18,437 –	5,131 KB	8,000 K

“Arial Unicode MS”
500
150K

フォントで使用するメモリの調整

1. UI
Flash
(

2. GFx::FontLib
( )
SWF/GFX
GFx::FontLib

3.
IME

4. GFxEport -fc
10 30% 5.3

GFx::FontLib
IME

1.2 ゲーム *UI* のフォントに関する決定

Scaleform UI

4

1:

2:

3:

Scaleform

4:

Scaleform

UI

UI

2

3

読み込みフォントの置き換え Player

GFx::FontLib

デバイスフォントエミュレーション

( )

カスタムアセットの生成 Flash SWF/GFX

Scaleform

3: Scaleform

GFx::FontLib

FreeType 2

GFx::FontLib

4 Scaleform

3 gfxexport

パート　ゲームフォントライブラリの作成

UI

UI

UI

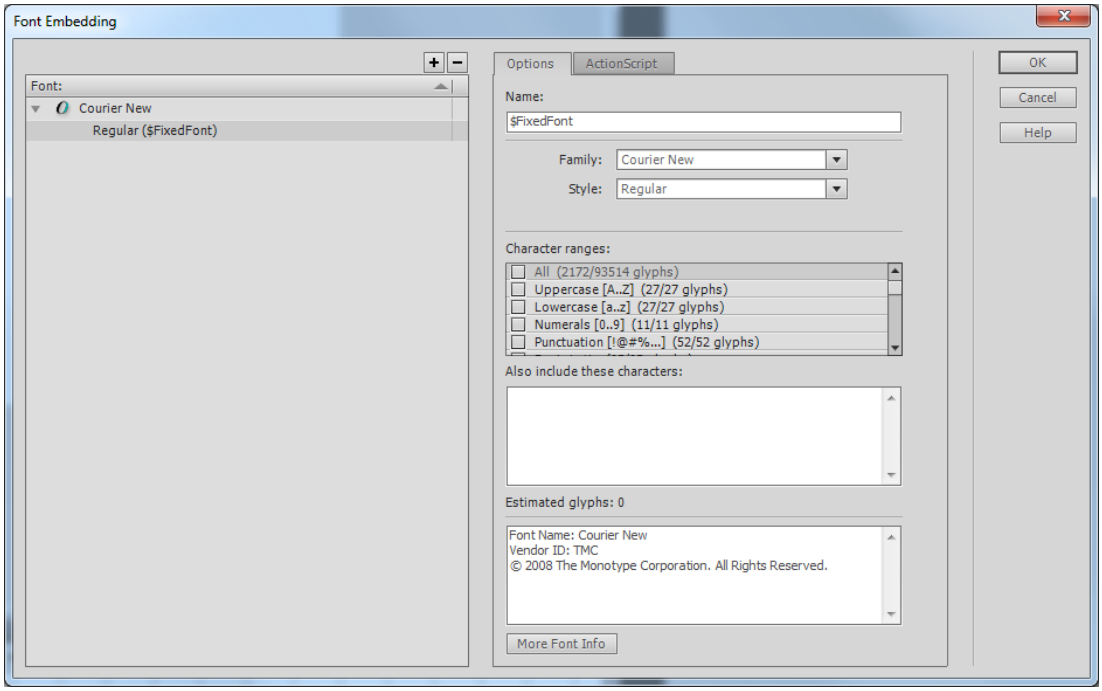
UI

Scaleform

fonts_en.swf　fonts_kr.swf　Flash
SWF
Flash

2.1 フォントシンボル

Flash Studio
W　L
[　…]



FLA

“$FixedFont”

“$ArialGame*”

UI

‘$’

Web

‘_’

-

[

]

Flash Studio

$

フォントシンボルの書き出しと読み込み

FLA

/

FLA

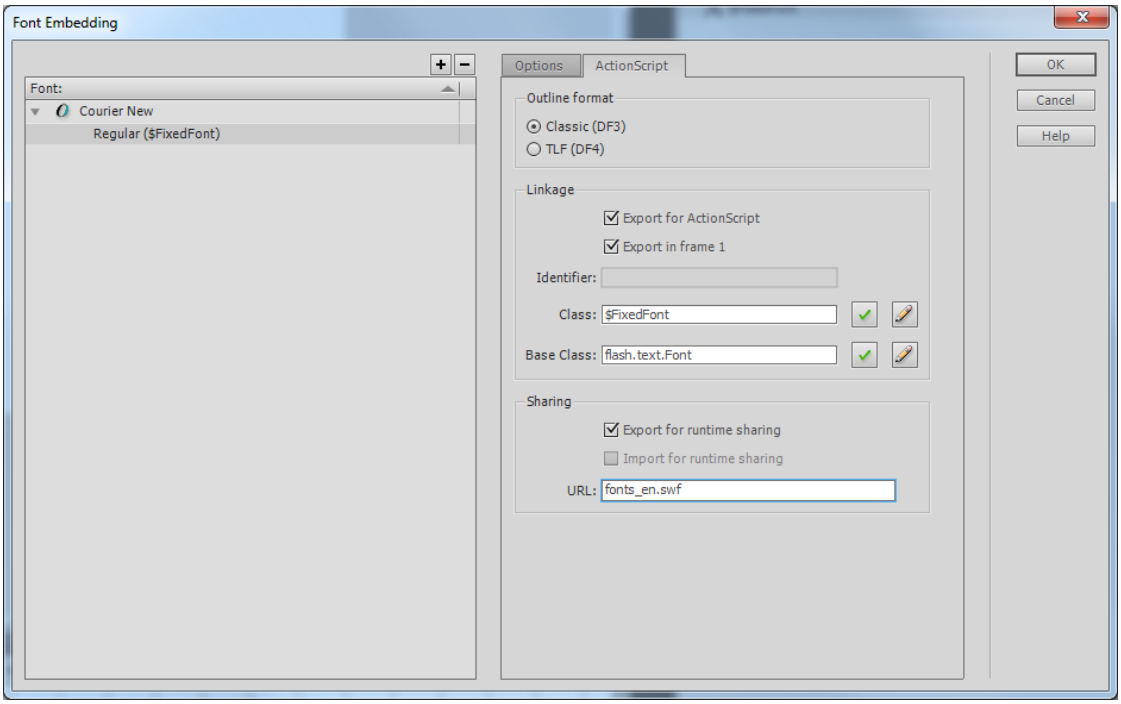
[ ]

FLA

[ ]

FLA

Flash /
"Properties"
"ActionScript"



"OK" Flash

ActionScript
URL
SWF
GFx::FontLib URL

"fonts_en.swf" **のデフォルトでのフォントライブラリがフォントの置換を使用するように設定する必要があります。**

```
Loader.SetDefaultFontLibName("fonts_en.swf");
```

GFxPlayer fonconfig.txt
`fontlib "fonts_en.swf"`

fontlib

“ : $ ”
/

SWF
Scaleform SWF

TextField.htmlText TextFormat
(“Arial”)
ActionScript
( )

書き出されたフォントと文字の埋め込み

Flash Studio CS4

Windows [ ] > [ ] > [Unicode] “English”
243 11,920
"gfxfontlib.swf"

Flash CS5
"gfxfontlib.swf"
fonts_en.swf fonts_jp.swf

Flash Studio CS4
"gfxfontlib.swf"
GFx 4.0

2.2 gfxfontlib.swf を作成する手順

Flash Studio FLA

1. ‘$TitleFont’
‘$NormalFont’
2. FLA
3. 1

4.

[ ]

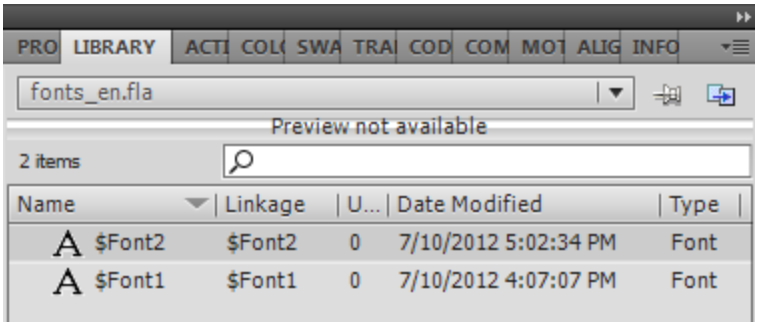
5.

6. fonts_<lang>.swf lang

‘en’ ‘ru’ ‘jp’ ‘cn’ ‘kr’

FontMap

FLA



/

パート　多国語化のアプローチの選択

UI

7. 読み込みフォントの置き換え
8. デバイスフォントエミュレーション
9. カスタムアセットの生成

GFx::FontLib　GFx::FontMap　‘fonts_en.swf’

GFx::Translator

GFx::Translator　Flash Studio
UI
UI

3.1 読み込みフォントの置き換え

Flash
/
1　fonts_en.swf
UI

UI　UI　Adobe Flash Player　“fonts_en.swf”

Scaleform

Loader::SetDefaultFontLibName (const char* filename)　　C++
‘filename’

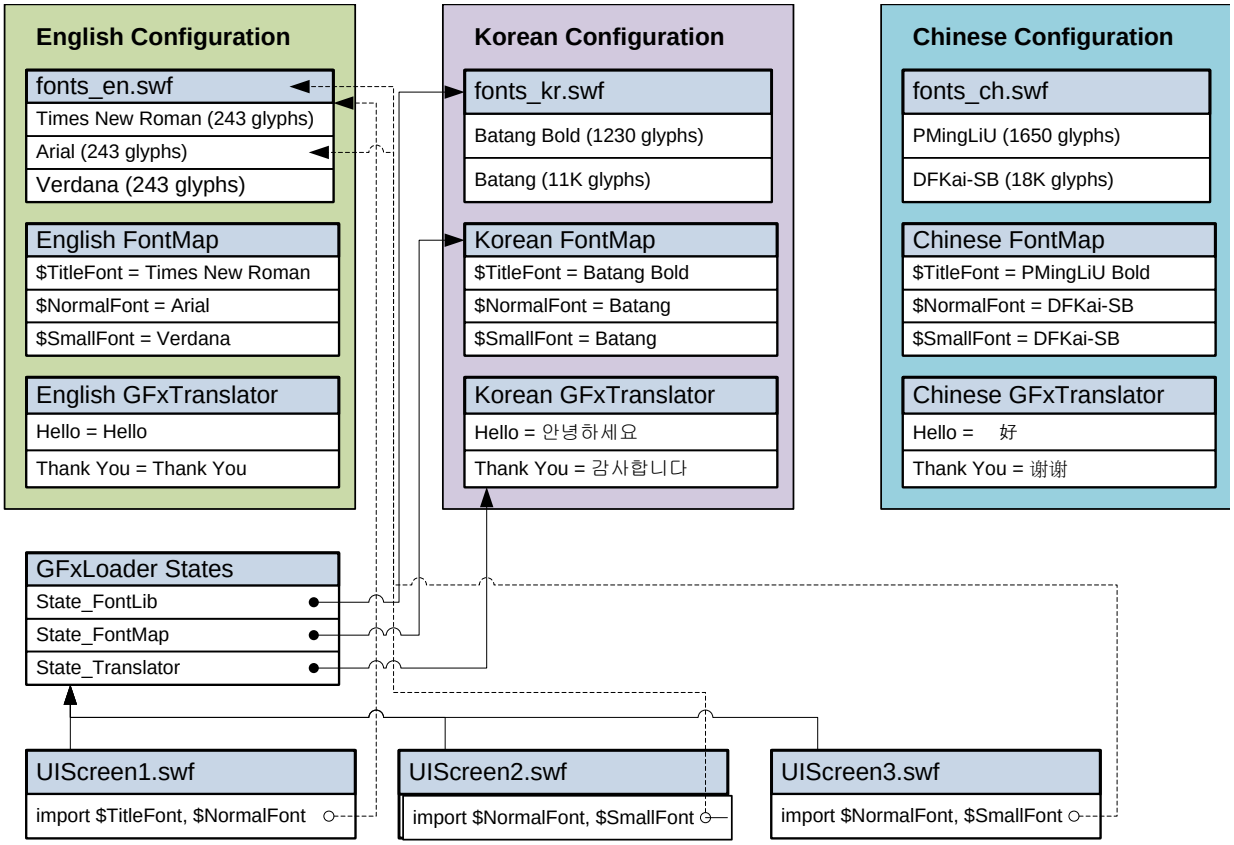
```
Loader.SetDefaultFontLibName(“fonts_kr.swf”);
```

SWF　　CreateInstance

1. 文字列変換テーブル: ゲームの各フレーズを翻訳されたフレーズにマップするものです。
2. :
3. : (“$NormalFont”)

Scaleform　　GFx::Loader
GFx::Translator　GFx::FontLib　　GFx::FontMap
GFx::Loader::CreateMovie

CreateMovie



“UIScreen1.swf” “UIScreen3.swf” “fonts_en.swf”

Scaleform GFx::Loader
GFx::FontLib GFx::FontMap
2 1
Loader::SetDefaultFontLibName() 2
GFx::FontLib UI

“fonts_kr.swf”

UI

多国語のテキストの設定

SWF
GFX
GFx::FontMap GFx::Translator

Scaleform “fontconfig.txt”

1. 1 “fonts_en.swf”
2. GFx::Translator
3. GFx::FontMap
4. “fonts_kr.swf”

UI

1. GFx::Loader GFx::Translator Translate
2. GFx::FontMap GFx::Loader MapFont GFx::FontMap GFx::FontMap 3
3. (GFx::FontCacheMamanger, GFx::FontProvider GFx::FontPackParams)
4. GFx::FontLib GFx::Loader
5. SetDefaultFontLibName(filename) swf
   `Loader.SetDefaultFontLibName(“fonts_en.swf”);`

   GFx::FontLib SWF/GFX GFx::FontLib::AddFromFile
6. GFx::Loader::CreateMovie

上での多国語化

Scaleform Player (
“fontconfig.txt”) SWF/GFX Scaleform Player
‘fontconfig.txt’
/fc
Ctrl+N
F2 Scaleform Player HUD

8 ASCII UTF-16

```
[FontConfig "English"]
fontlib "fonts_en.swf"
map "$TitleFont" = "Times New Roman" Normal
map "$NormalFont" = "Arial " Normal
map "$SmallFont" = "Verdana" Normal

[FontConfig "Korean"]
fontlib "fonts_kr.swf"
map "$TitleFont" = "Batang" Bold
map "$NormalFont" = "Batang" Normal
map "$SmallFont" = "Batang" Normal
tr  "Hello" = "안녕하세요"
tr  "Thank You" = "감사합니다"

[FontConfig "Chinese"]
fontlib "fonts_ch.swf"
map "$TitleFont" = "PMingLiU" Bold
map "$NormalFont" = "DFKai-SB" Normal
map "$SmallFont" = "DFKai-SB" Normal
tr  "Hello" = "你  "
tr  "Thank You" = "谢谢"
```

`[FontConfig "name"]` Scaleform Player HUD

設定ステートメント	意味
`fontlib "fontfile"`	SWF/GFX GFx::FontLib fontlib
`map "$UIFont" = "PMingLiU"`	GFx::FontMap GFx::FontMap UI `$UIFont` “fonts_en.swf” UI

tr "Hello" = "你 "	

Scaleform

Scaleform XML

FontConfigParser.h/.cpp

Scaleform
Bin\Data\AS2\Samples\FontConfig and Bin¥Data¥AS3¥Samples¥FontConfig
Scaleform Player “sample.swf”
“fontconfig.txt”

デバイスフォントエミュレーション

GFx::FontMap
UI (“Arial” “Verdana”)
UI Scaleform Player

“Arial”
“Batang” “Batang” GFx::FontLib
“font_kr.swf”

GFx::FontProviderWin32
GFx::FontProviderFT2
GFx::FontLib

Flash Adobe Flash Player

Scaleform Flash
UI
( )
Scaleform Player
(ActionScript TextField.antiAliasType
)
UI
Scaleform Adobe Flash
Player
lib “fonts_en.swf”

ActionScript TextFormat Flash
(“$TitleFont”)
(“Arial”)

“fonts_en.swf”
“fonts_en.swf”

”fonts_en.swf” UI

フォントライブラリ作成の詳細手順

2 fontconfig.txt SWF

1. FLA ActionScript 2.0 3.0 ‘main-app.fla’
SWF
2. FLA ActionScript
‘fonts_en.swf’
3. ‘Library’ ‘New Font…’
$Font1 “Family” “Arial” “Style” “Regular”
4. “Character ranges” 'Basic Latin'

5. ‘ActionScript’ ‘Export for ActionScript’ ‘Export for frame 1’ ‘Export for run-time sharing’ ‘URL’ ‘fonts_en.swf’
OK ActionScript 3.0 Flash Studio ‘A definition for this class could not be found…’ ‘OK’

6. 3 5 ‘$Font2’ ‘Family’ /
‘Style’ ‘Family’ ‘Style’

8. ‘Library’ ‘New Font...’
   $Font1 “Family” “ ” “Style”
   “Regular”
9. “Character ranges” ‘Korean Hangul
   (All)’
10. ‘ActionScript’ ‘Export for ActionScript’ ‘Export for frame 1’ ‘Export
   for run-time sharing’ ‘URL’ ‘fonts_kr.swf’
   OK ActionScript 3.0 Flash Studio ‘A definition
   for this class could not be found...’ ‘OK’

11. 8 10 ‘$Font2’ ‘Family’
   ‘ ’
12. ‘fonts_en.swf’ ‘fonts_kr.swf’
13. main-app.fla.
   ‘fonts_en.swf’ ‘$Font1’ ‘$Font2’
   Ctrl-C ‘Copy’
14. ‘main-app.fla’ j Ctrl-V
   ‘Paste’ ‘Import:’

15. ‘main-app.fla’ ‘Classic Text’ ‘Dynamic Text’
   ‘Family’ ‘$Font1*’
   ‘$TEXT1’ ID
16. 2 ‘$Font2*’ ‘$TEXT2’
17. ‘main-app.swf’
18. swf fontconfig.txt Unicode
   UTF-8

```
[FontConfig "English"]
fontlib "fonts_en.swf"
map "$Font1" = "Arial"
map "$Font2" = "Times New Roman"
tr  "$TEXT1" = "This is"
tr  "$TEXT2" = "ENGLISH!"


[FontConfig "Korean"]
fontlib "fonts_kr.swf"
map "$Font1" = "Batang"
map "$Font2" = "BatangChe"
tr  "$TEXT1" = "이것"
tr  "$TEXT2" = "은 한국이다!"
```

“ ” “ ” Flash SWF

“Batang” “BatangChe” fontconfig.txt

SWF

gfxexport –fntlst <swfname> *.lst

19. Unicode UTF-8

GFxPlayer ‘main-app.swf’ ‘This is’ ‘English!’ Ctrl-N fonts_kr.swf

fonts_kr.swf

3.2 カスタムアセットの生成

SWF/GFX

“UIScreen.fla”

SWF

“UIScreen.swf” “UIScreen.swf”

(UI 600K

)

UI

UI (
)

UI IME

Flash Studio
[
] (Ctrl + F11)
Scaleform Flash [ ]

パート　フォントソースの設定

Flash
HTML

1. ローカルに埋め込まれたフォント
2. SWF/GFX
3. GFx::FontLib SWF/GFX
4. (GFx::FontProviderWin32)

Flash

GFx::FontLib
GFx::FontMap GFx::FontProvider GFx::FontLib GFx::FontMap
SWF/GFX

GFx::Loader::SetFontProvider
SWF
SWF
Scaleform

GFx::FontProviderWin32 – Win32 API
GFx::FontProviderFT2 – David Turner FreeType-2

4.1 フォントの参照順

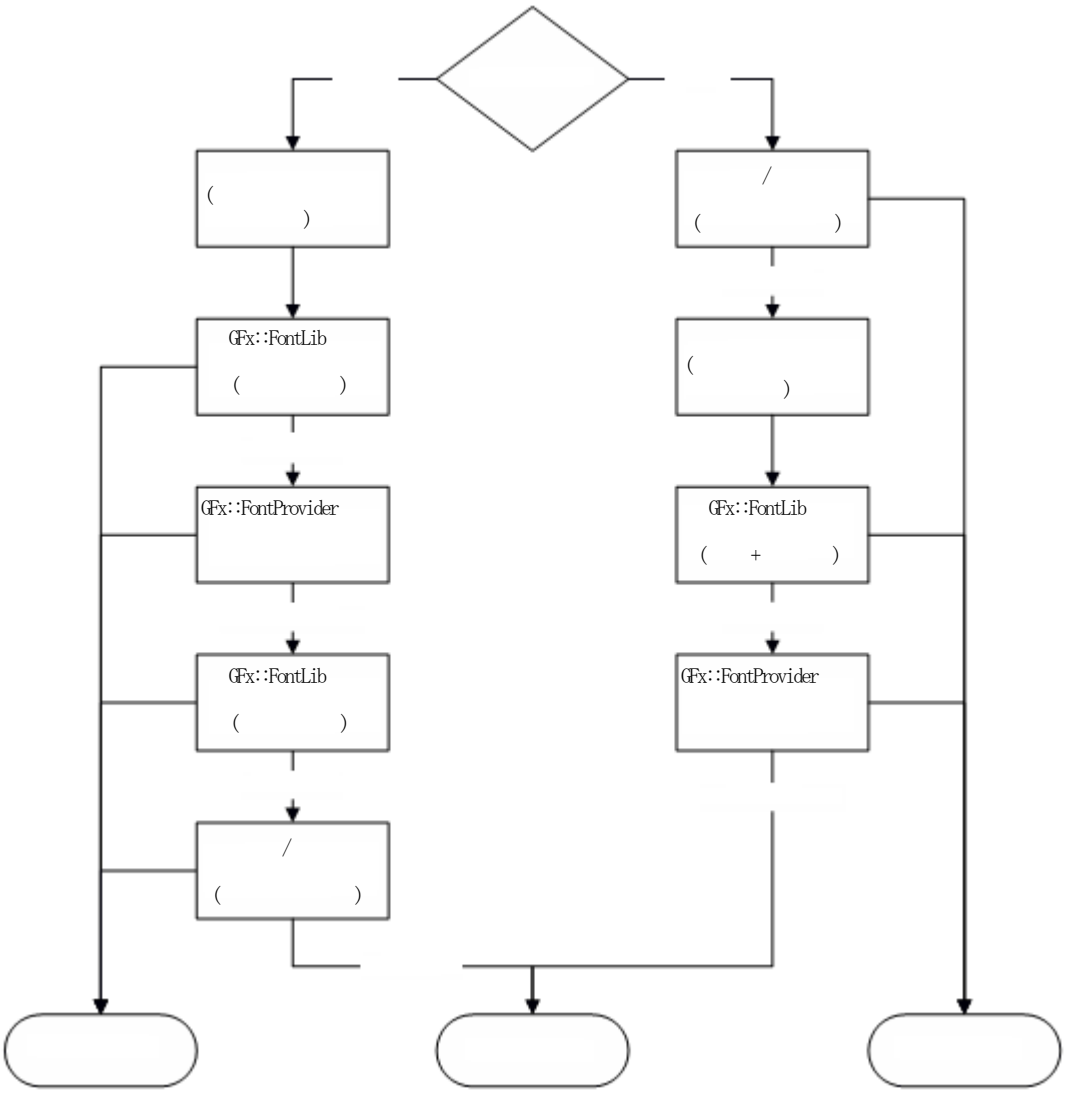
Scaleform
[
]

Adobe Flash

(-

) Scaleform

( )



GFx::FontLib GFx::FontProvider

4.2 GFx::FontMap

GFx::FontMap

3

```
[FontConfig "Korean"]
fontlib "fonts_kr.swf"
map "$TitleFont" = "Batang" Bold
map "$NormalFont" = "Batang" Normal
map "$SmallFont" = "Batang" Normal
```

(“$TitleFont”)
( )

C++

```
#include "GFx/GFx_FontLib.h"
. . .
Ptr<FontMap> pfontMap = *new FontMap;
Loader.SetFontMap(pfontMap);

pfontMap->MapFont("$TitleFont", "Batang", FontMap:: MFF_Bold,scaleFactor = 1.0f);
pfontMap->MapFont("$NormalFont", "Batang", FontMap:: MFF_Normal,scaleFactor = 1.0f);
pfontMap->MapFont("$SmallFont", "Batang", FontMap:: MFF_Normal,scaleFactor = 1.0f);
```

“$NormalFont”
“$SmallFont”
3 MapFont
MFF_Original

scaleFactor
scaleFactor 1.0f

GFx::FontMap

GFx::Loader::CreateMovie
GFx::MovieDef

4.3 GFx::FontLib

3 GFx::FontLib 1 “fonts_en.swf” 2

```
#include "GFx/GFx_FontLib.h"
. . .
Ptr<FontLib> fontLib = *new FontLib;
Loader.SetFontLib(fontLib);
fontLib->SetSubstitute(“<default_font_lib_swf_or_gfx_file>”);

Ptr<MovieDef>  m1 = *Loader.CreateMovie("<swf_or_gfx_file1>");
Ptr<MovieDef>  m2 = *Loader.CreateMovie("<swf_or_gfx_file2>");
. . .
fontLib->AddFontsFrom(m1, true);
fontLib->AddFontsFrom(m2, true);
```

AddFontsFrom AddFontsFrom 2 AddRef (m1 m2) (false) /

GFx::FontLib

4.4 GFx::FontProviderWin32

GFx::FontProviderWin32 Win32 API GetGlyphOutline()

```
#include "GFx/GFx_FontProviderWin32.h"
. . .
Ptr<FontProviderWin32> fontProvider = *new FontProviderWin32(::GetDC(0));
Loader.SetFontProvider(fontProvider);
```

GFx::FontProviderWin32 Windows (DC) DC (::GetDC(0))

ネイティブにヒントの付いたテキストを使う

Scaleform
CJK (Chinese / Japanese / Korean)
CJK
CJK
API
GFx::FontProviderWin32 GFx::FontProviderFT2

4

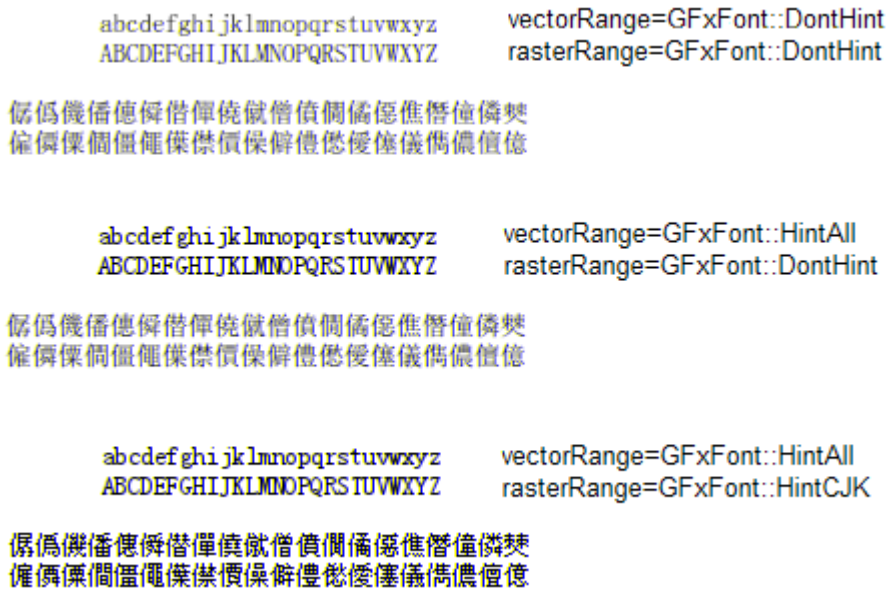
```
Font::NativeHintingRange vectorRange;
Font::NativeHintingRange rasterRange;
unsigned maxVectorHintedSize;
unsigned maxRasterHintedSize;
```

vectorRange rasterRange

```
Font::DontHint -
Font::HintCJK  -
Font::HintAll  -
```

rasterRange vectorRange maxVectorHintedSize maxRasterHintedSize

SimSun



“Arial Unicode MS “

abcdefghijklmnopqrstuvwxyz
ABCDEFGHIJKLMNOPQRSTUVWXYZ
僞僞儀僠憓僞僭僤僤僤僥僦僧僨僩僑僊僭僮僰
僱僾僳僧僵僶僕僭僨僦僭僭僱僱僱僱僱僱

Adobe FLASH Scaleform

Win32 FreeeType

```
vectorRange = Font::DontHint;
rasterRange = Font::HintCJK;
maxVectorHintedSize = 24;
maxRasterHintedSize = 24;
```

ネイティブヒンティングの設定

GFx::FontProviderWin32 CJK

```
fontProvider->SetHintingAllFonts(. . .);
```

or

```
fontProvider->SetHinting(fontName, . . .);
```

SetHintingAllFonts ( ) SetHinting( ) ( )

```
void SetHintingAllFonts(Font::NativeHintingRange vectorRange,
                        Font::NativeHintingRange rasterRange,
                        unsigned maxVectorHintedSize=24,
                        unsigned maxRasterHintedSize=24);
```

```
void SetHinting(const char* Name,
                Font::NativeHintingRange vectorRange,
                Font::NativeHintingRange rasterRange,
                unsigned maxVectorHintedSize=24,
                unsigned maxRasterHintedSize=24);
```

Name UTF-8

4.5 GFx::FontProviderFT2

David Turner FreeType-2

GFx::FontProviderWin32

FreeType

( )

Windows MSVC

freetype<ver>.lib – DLL
freetype<ver>_D.lib – DLL
freetype<ver>MT.lib – (CRT)
freetype<ver>MT_D.lib – (CRT)

“<ver>” FreeType (2.1.9 219)

freetype2/include freetype2/objs

GFx::FontProviderFT2

```
#include "GFx/GFx_FontProviderFT2.h"
. . .
Ptr<FontProviderFT2> fontProvider = *new FontProviderFT2;
<Map Font to Files or Memory>
Loader.SetFontProvider(fontProvider);
```

```
FontProviderFT2(FT_Library lib=0);
```

FreeType ( )
FreeType FreeType Scaleform

(FT_Done_FreeType)

GFx::Loader FreeType
( )

fontProvider->GetFT_Library() FT_Library
FreeType
GFx::FontProviderFT2 FreeType

フォントのファイルへのマッピング

Win32 API FreeType ( )

GFx::FontProviderFT2

```
void MapFontToFile(const char* fontName, unsigned fontFlags,
                   const char* fileName, unsigned faceIndex=0
                   Font::NativeHintingRange vectorHintingRange = Font::DontHint,
                   Font::NativeHintingRange rasterHintingRange = Font::HintCJK,
                   unsigned maxVectorHintedSize=24,
                   unsigned maxRasterHintedSize=24);
```

“fontName” (“Times New Roman”)
“fileName” (“C:\\WINDOWS\\Fonts\\times.ttf”)
“fontFlags” “Font::FF_Bold” “Font::FF_Italic” “Font::FF_BoldItalic”
“Font::FF_BoldItalic”
“Font::FF_Bold | Font::FF_Italic” “faceIndex” FT_New_Face
0

Win32 FreeType

フォントのメモリへのマッピング

FreeType

```
void MapFontToMemory(const char* fontName, unsigned fontFlags,
                     const char* fontData, unsigned dataSize, unsigned faceIndex=0
                     Font::NativeHintingRange vectorHintingRange = Font::DontHint,
                     Font::NativeHintingRange rasterHintingRange = Font::HintCJK,
                     unsigned maxVectorHintedSize=24,
                     unsigned maxRasterHintedSize=24);
```

```
FILE* fd = fopen("C:\\WINDOWS\\Fonts\\times.ttf", "rb");
if (fd) {
    fseek(fd, 0, SEEK_END);
    unsigned size = ftell(fd);
    fseek(fd, 0, SEEK_SET);
    char* font = (char*)malloc(size);
    fread(font, size, 1, fd);
    fclose(fd);
    fontProvider->MapFontToMemory("Times New Roman", 0, font, size);
}
```

( )

GFx::Loader

FreeType Scaleform

Windows FreeType

```
Ptr<FontProviderFT2> fontProvider = *new FontProviderFT2;
fontProvider->MapFontToFile("Times New Roman", 0,
                            "C:\\WINDOWS\\Fonts\\times.ttf");
fontProvider->MapFontToFile("Times New Roman", Font::FF_Bold,
                            "C:\\WINDOWS\\Fonts\\timesbd.ttf");
fontProvider->MapFontToFile("Times New Roman", Font::FF_Italic,
                            "C:\\WINDOWS\\Fonts\\timesi.ttf");
fontProvider->MapFontToFile("Times New Roman", Font::FF_BoldItalic,
                            "C:\\WINDOWS\\Fonts\\timesbi.ttf");

fontProvider->MapFontToFile("Arial", 0,
                            "C:\\WINDOWS\\Fonts\\arial.ttf");
fontProvider->MapFontToFile("Arial", Font::FF_Bold,
                            "C:\\WINDOWS\\Fonts\\arialbd.ttf");
fontProvider->MapFontToFile("Arial", Font::FF_Italic,
                            "C:\\WINDOWS\\Fonts\\ariali.ttf");
fontProvider->MapFontToFile("Arial", Font::FF_BoldItalic,
                            "C:\\WINDOWS\\Fonts\\arialbi.ttf");

fontProvider->MapFontToFile("Verdana", 0,
                            "C:\\WINDOWS\\Fonts\\verdana.ttf");
fontProvider->MapFontToFile("Verdana", Font::FF_Bold,
                            "C:\\WINDOWS\\Fonts\\verdanab.ttf");
fontProvider->MapFontToFile("Verdana",Font::FF_Italic,
                            "C:\\WINDOWS\\Fonts\\verdanai.ttf");
fontProvider->MapFontToFile("Verdana", Font::FF_BoldItalic,
```

```
                              "C:\\WINDOWS\\Fonts\\verdanaz.ttf");
. . .
Loader.SetFontProvider(fontProvider);
```

MapFontToFile()

パート　フォント レンダリングの設定

Scaleform
(1　2　)

1.

2. (　mip-map
)　gfxexport

3. (　GFx::FontPackParams
)

4.

PC　Xbox 360　PS3　Wii

GFx::FontProviderWin32　GFx::FontProviderFT2

gfxexport　CPU
(PSP　PS2　)

Scaleform

5.1 フォントキャッシュマネージャ

Scaleform
Render::GlyphCacheConfig GFx::FontPackParams
Renderer2D

NULL

GFX

アップグレードメモ
Scaleform 3.x 4.0
GFx 3.3 GFxLoader GFxFontCacheManager
GFx 4.0 Renderer2D
GlyphCacheConfig Scaleform GFx 4.0
5.2

GFx::FontPackParams ‘gfxexport’

( )

```
bool Done = false;

if (Font has Static Textures with Packed Glyphs)
{
    if (GlyphSize <
        FontPackParams.TextureConfig.NominalSize * MaxRasterScale)
    {
        Draw the Glyph as a Texture using Static Cache;
        Done = true;
```

```
        }
    }
    else if (Dynamic Cache is Enabled AND
             GlyphSize < GlyphCacheParams.MaxSlotHeight)
    {
        Draw the Glyph as a Texture using Dynamic Cache;
        Done = True;
     }

    if (Not Done)
    {
        Draw the Glyph as Vector Shape;
    }
```

5.2 ダイナミックフォントキャッシュの使用

(　　10000
)

```
renderer->GetGlyphCacheConfig()->SetParams(Render::GlyphCacheParams(0));
```

renderer　　Render::Renderer2D

HAL

LRU (Least Recently Used)

```
Render::GlyphCacheParams gcparams;
gcparams.TextureWidth   = 1024;
gcparams.TextureHeight  = 1024;
gcparams.MaxNumTextures = 1;
gcparams.MaxSlotHeight  = 48;
gcparams.SlotPadding    = 2;
gcparams.TexUpdWidth    = 256;
gcparams.TexUpdHeight   = 512;

renderer->GetGlyphCacheConfig()->SetParams(gcparams);
```

TextureWidth, TextureHeight –
2

MaxNumTextures –

MaxSlotHeight –

SlotPadding –
2

TexUpdWidth, TexUpdWidth –
256x512 (128K) 256x256 128x128

LRU
1MB 4KB
4 K
/ 256
4 K 16
100 256 4K
4K 1 MB

256
2
MaxSlotHeight+2*SlotPadding 2

2-5
(
) 60-80%

( )
UI 500 2000
1

1

(
)
(
)
Flash

GFX, Dynamic cache	Readability	Animation
Anti-alias for readability *Anti-alias for animation*	A	A

(
)
(
)
Scaleform
Flash

Scaleform
(O G C Q o g e)

Scaleform

H E F T U V W X Z
z x v w

1
1
Scaleform

Flash
[
]
“Zz”

5.3 フォントのコンパクター、*gfxexport* の使用

gfxexport SWF
GFX SWF
DDS TGA
-fc gfxeport

コマンドラインのオプション	ビヘイビア
-fc	
-fcl <size>	256 256 25% Flash 1024
-fcm	FontCompactor 10 70% 12 32 16 64 12 32 16 64

5.4 フォント テクスチャの前処理 - *gfxexport*

gfxexport –fonts
Scaleform Player GFX

普通の x アプリケーションにはエクスポートされたテクスチャの使用は推奨しません。

gfxexport

CPU

GFx::ImageCreator

Render::Renderer

mip-map ( )

‘test.swf’ ‘test.gfx’

```
gfxexport –fonts –strip_font_shapes test.swf
```

strip_font_shapes Scaleform

gfxexport

gfxexport

コマンドライン オプション	**動作**
-fonts	( )
-fns <size>	( ) 48 1 mip-map
-fpp <n>	3
-fts <WxH>	256x256 1 ('-fts 128' 128x128) '-fts 512x128'
-fs	

コマンドライン オプション	動作
- strip_font_shapes	GFX
-fi <format>	<format> TGA8 (grayscaled) TGA24 (grayscaled) TGA32 DDS8 (-i) TGA TGA8 DDS A8

警告

5.2

5.5 フォント グリフ パッカーの設定

GFxExport

( )

Scaleform

```
Ptr<FontPackParams> packParams = *new FontPackParams();
Loader.SetFontPackParams(packParams);
```

GFxExport (.Scaleform)

GFxExport

( )

```
Loader.GetFontPackParams()->SetUseSeparateTextures(Bool flag);
Loader.GetFontPackParams()->SetGlyphCountLimit(int lim);
Loader.GetFontPackParams()->SetTextureConfig(fontPackConfig);
```

`SetUseSeparateTextures()` "true"

"false"

SetGlyphCountLimit() 0 ( )

500

( )

SetTextureConfig()

```
FontPackParams::TextureConfig fontPackConfig;
fontPackConfig.NominalSize   = 48;
fontPackConfig.PadPixels     = 3;
fontPackConfig.TextureWidth  = 1024;
fontPackConfig.TextureHeight = 1024;
```

NominalSize– ( )

RAM

“NominalSize”

NominalSize ( )

PadPixels – 1

TextureWidth TextureHeight – 2

1

.GFX

2
```
Ptr<FontPackParams> packParams = *new FontPackParams();
Loader.SetFontPackParams(packParams);
...
renderer->GetGlyphCacheConfig()->SetParams(Render::GlyphCacheParams(0));
```

3
```
Ptr<FontPackParams> packParams = *new
FontPackParams();Loader.SetFontPackParams(packParams);
Loader.GetFontCacheManager()->EnableDynamicCache(false);
```

500

( )

5.6 ベクトル化の制御

Scaleform

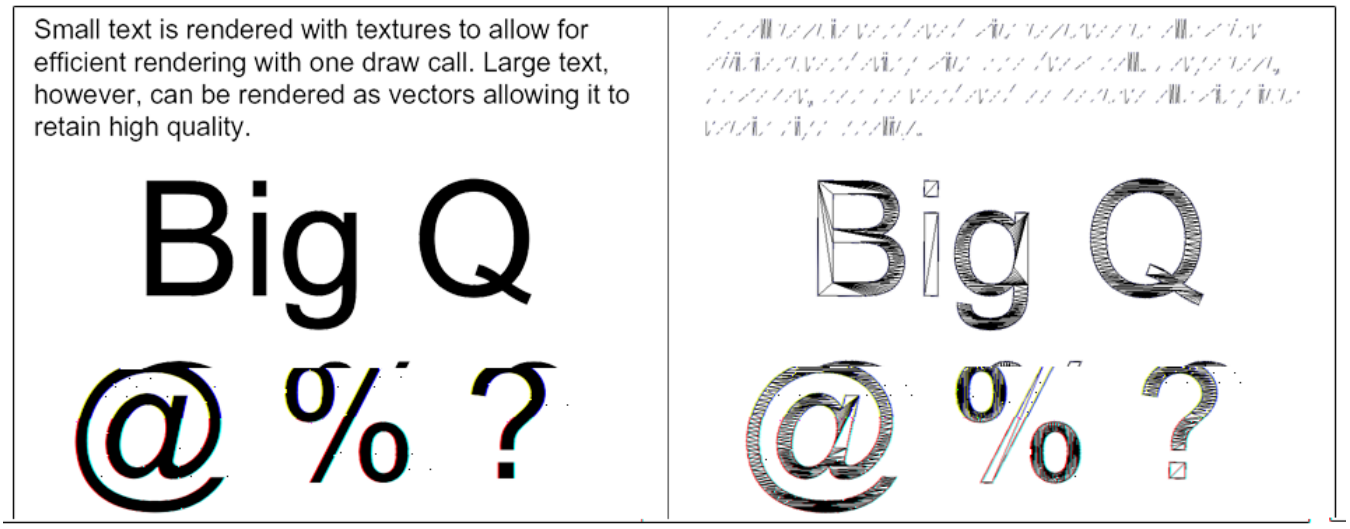
Flash

1

Scaleform

Scaleform Player

Ctrl+W



(MaxSlotHeight) Render::GlyphCacheConfig::SetParams
MaxSlotHeight

1

(mip-maps)

GlyphCacheParams:: MaxRasterScale
SetMaxRasterScale ( )

MaxRasterScale 1 1.25
1.25 Player
48
60

MaxRasterScale
strip_font_shapes gfxexport
GFX

パート　テキストのフィルタエフェクトと　拡張

Scaleform

Player

Scaleform　Flash
Adobe Flash

Flash
Scaleform

Scaleform

6.1 フィルタの種類、使用可能なオプションと制約

Flash Studio　1

Flash　Scaleform　Scaleform
Flash

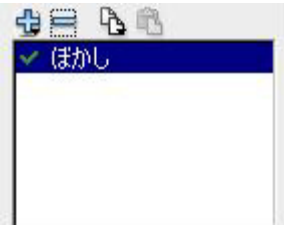
Scaleform
Flash
Scaleform

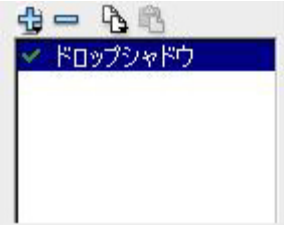
-
-

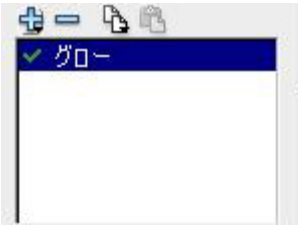
-

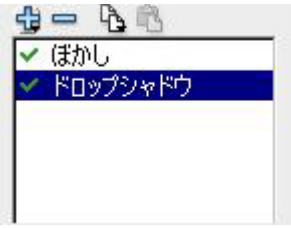
-

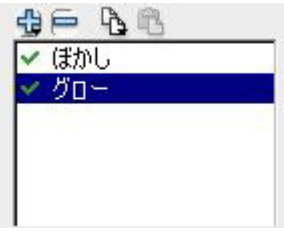
-











Flash

[ ] [ ]
(2) Scaleform
[( )] [( )]

Scaleform
Blur X Blur Y 15.9
1590%

[ ] Flash
Scaleform

Flash GFx

[ ] (
)

6.2 フィルタ品質

Flash [ ]
[ ] [ ]
3x3 3x3 [ ] [ ]
2 [ ] 3 [ ]

Flash Scaleform
Scaleform [ ]
[ ] [:]

( )
Scaleform
(
)

	Low quality	Medium quality	High quality
Flash	Text	Text	Text
GFx	Text	Text	Text

Flash Scaleform Scaleform

http://www.ph.tn.tudelft.nl/Courses/FIP/noframes/fip-Smoothin.html

6.3 フィルタのアニメーション

Flash Scaleform ActionScript

Scaleform

( )

( )

6.4 ActionScript からのフィルターの使用

Scaleform AS2 AS3 /
DropShadowFilter BlurFilter ColorMatrixFilter BevelFilter
Flash