cocos2d-x中有三个雷可以在层或精灵中添加文字:

**LabelTTF LabelBMFont LabelAtlas**

LabelTTF直接支持TTF字体，可以支持全部的中文，但效率稍低。

LabelBMFont适合显示特定的文字，通过预先将文字生成图片，提高了效率，但是不能支 持全部中文

LabelAtlas适合使用的文字不多，组合多，但是文字的编码是连续的，比如数字或者英文字符

**LabelTTF**

**Label\* \_renderLabel;**

bool \_contentDirty;

FontDefinition \_fontDef;

实际上LabelTTF保存中了一个Label，并且这个Label作为该LabelTTF的子节点，查看LabelTTF中的构造函数：

**LabelTTF::LabelTTF()**

{

\_renderLabel = Label::create();

\_renderLabel->setAnchorPoint(Vec2::ANCHOR\_BOTTOM\_LEFT);

this->addChild(\_renderLabel);

this->setAnchorPoint(Vec2::ANCHOR\_MIDDLE);

\_contentDirty = false;

\_cascadeColorEnabled = true;

\_cascadeOpacityEnabled = true;

}

查看LabelTTF的visit函数：

void LabelTTF::visit(Renderer \*renderer, const Mat4 &parentTransform, uint32\_t parentFlags)

{

if (\_contentDirty)

{

this->setContentSize(\_renderLabel->getContentSize());

\_contentDirty = false;

}

Node::visit(renderer,parentTransform, parentFlags);

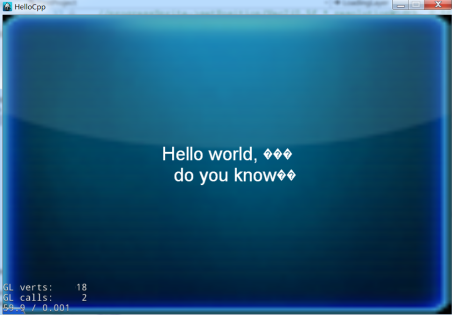
}

实际上这里操作的都是Label，这里获取的ContentSize也是Label的ContentSize，Node：：Visit中，因为Label作为LabelTTF的子节点，因此也最终会调用Label的draw函数。

LabelTTF \*label1 = LabelTTF::create("Hello world, 你好\n do you know！", "arial.ttf", 40);

label1->setPosition(Vec2(0.5f \* resolutionWidth, 0.5f \* resolutionHeight));

addChild(label1);



**LabelBMFont**

std::string \_fntFile;

Label\* \_label;

实际上LabelBMFont也是保存了一个Label，并且这个Label作为LabelBMFont的子节点，查看其构造函数：

LabelBMFont::LabelBMFont()

{

\_label = Label::create();

\_label->setAnchorPoint(Vec2::ANCHOR\_BOTTOM\_LEFT);

this->addChild(\_label);

this->setAnchorPoint(Vec2::ANCHOR\_MIDDLE);

\_cascadeOpacityEnabled = true;

#if CC\_LABELBMFONT\_DEBUG\_DRAW

\_debugDrawNode = DrawNode::create();

addChild(\_debugDrawNode);

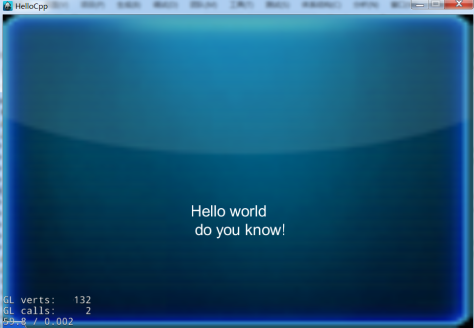
#endif

}

LabelBMFont \*label2 = LabelBMFont::create("Hello world \n do you know!", "font.fnt");

label2->setPosition(Vec2(0.5f \* resolutionWidth, 0.5f \* resolutionHeight - 100.0f));

addChild(label2);



**终极大boss：Label**

首先查看一个结构体TTFConfig

typedef struct \_ttfConfig

{

std::string fontFilePath;

float fontSize;

GlyphCollection glyphs;

const char \*customGlyphs;

bool distanceFieldEnabled;

int outlineSize;

\_ttfConfig(const std::string& filePath = "",float size = 12, const GlyphCollection& glyphCollection = GlyphCollection::DYNAMIC,

const char \*customGlyphCollection = nullptr, bool useDistanceField = false, int outline = 0)

: fontFilePath(filePath)

, fontSize(size)

, glyphs(glyphCollection)

, customGlyphs(customGlyphCollection)

, distanceFieldEnabled(useDistanceField)

, outlineSize(outline)

{

if(outline > 0)

{

distanceFieldEnabled = false;

}

}

} TTFConfig;

cocos2dx中的简单说明：

\* Label can be created with:

\* - A true type font file.

\* - A bitmap font file.

\* - A char map file.

\* - The built in system font.