

# 分析FBReader源码(2) --- 文件结构及基本框架梳理

创建时间: 2012/05/15 17:56

更新时间: 2012/05/15 20:59

作者: Nil

来源: 六阅雪:<http://sixreader.com>

## 一、编译源代码

紧接上篇文章，配置好环境后，接下来就是编译源代码运行。当初直接点击“运行”，但那是引用了它已经生成好的SO库的情况下，今后将开始简单修改代码（c/c++），所以会重新编译。过程如下：

1.找到FBReader所在的路径(类似)：

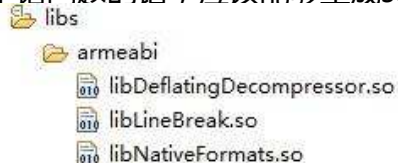
C:\Users\Luzaimou\Desktop\FBReader2.0\FBReaderJ

2.打开cygwin,跑到对应的目录：

```
luzaimou@SynWith /cygdrive/c/users/luzaimou/desktop/fbreader2.0/fbreaderj
$
```

3.直接输入 “\$NDK/ndk-build”即可，编译可能需要点时间。

4.不错问题的话，应该能够生成so文件了，刷新一下项目的libs目录。



5.运行。

## 二、FBReader文件结构

FBReader开源项目在国内还是挺流行的，但是目前貌似很少有这方面的资料，可能是高手都没时间，不愿意写。我很浅薄，冒天下之大不韪，希望各位高手不要笑话。写文章的初衷本就是秉承开源、共享、免费、自由的精神，所以也是跟随FBReader的本意，没有其他目的。文章的好坏，也是任人评说，仁者见仁，智者见智，只当抛砖引玉。望更多的童鞋加入其中，把开源的精神发扬广大！

我尽我所能写文章，如有纰漏，大家指点海涵（我也是刚刚研究2.0）。

FBReader的源码文件结构大致如下：

FBReader

_____src	( JAVA源代码 )
_____assets	( 这里有默认的那本书，还有背景图片等 )
_____docs	( 里面有一PDF,介绍如何安装Cygwin)
_____icons	( 一些图标，比如Menu,文字搜索的小图标 )
_____jni	(C/C++实现的库文件，内容比较多)
_____libs	( 生成的SO文件放这里 )
_____obj	( .a文件放这里 )
_____obsolete	( 用处不明 )
_____res	( 布局和string等xml放这里 )
_____scripts	( 用处不明 )
_____AndroidManifest.xml	

项目大概结构如此，下面介绍具体的Java部分的包。

具体内容如下：

com.nil.util	这个包是我自己写的，用于调试，不要管。
org.amse.ys.zip	这个包是解压用，epub本身就是一个压缩文件，所以要解压缩。
org.geometerplus.android.fbreader	这个包里，有一个主的Activity---FBReader.java,其他基本都是一些事件的Action
org.geometerplus.android.fbreader.api	这个包是FBReader通过AIDL实现的API接口，用于安装插件等
org.geometerplus.android.fbreader.crash	图书目录相关的修复，只有一个activity.
org.geometerplus.android.fbreader.image	长按某张图片，会调用包里的ImageViewActivity
org.geometerplus.android.fbreader.library	主要是初始化数据库books.db，还有图书信息的activity
org.geometerplus.android.fbreader.network	这里是FBReader网络连接相关的，比如下载书籍，书仓授权，OPDS相关。
org.geometerplus.android.fbreader.preferences	配置信息相关的，比如背景，文字大小，翻页动画速度，以及颜色等
org.geometerplus.android.fbreader.tree	一个BaseActivity和一个ListAdapter
org.geometerplus.android.util	三个工具类：包工具类，数据库，还有更新UI的
org.geometerplus.fbreader	里面只有一个文件，内容是整个FBReader项目的路径相关
org.geometerplus.fbreader.bookmodel	FBReader数据，会生成BookModel,里面有JAVA成返回Model和本地返回的Model
org.geometerplus.fbreader.fbreader	主要是图书显示的FBView,还有一些选择文字的 Action,翻页的Action
org.geometerplus.fbreader.formats	插件入口，FBReader把epub解析，txt解析等当做一个个插件
org.geometerplus.fbreader.formats.fb2	} 解析各种文件格式
org.geometerplus.fbreader.formats.html	
org.geometerplus.fbreader.formats.oeb	
org.geometerplus.fbreader.formats.pdb	
org.geometerplus.fbreader.formats.plucker	
org.geometerplus.fbreader.formats.util	} 网络相关
org.geometerplus.fbreader.formats.xhtml	
org.geometerplus.fbreader.library	
org.geometerplus.fbreader.network	
org.geometerplus.fbreader.network.atom	
org.geometerplus.fbreader.network.authentication	} 网络相关
org.geometerplus.fbreader.network.authentication.litres	
org.geometerplus.fbreader.network.opds	
org.geometerplus.fbreader.network.tree	
org.geometerplus.fbreader.network.urlInfo	
org.geometerplus.fbreader.tree	自定义的树结构，FBTree.
org.geometerplus.zlibrary.core.application	并不是Android里的Application,而是作者自定义的管理应用
org.geometerplus.zlibrary.core.config	内含一个抽象类，被SQLConfig实现。
org.geometerplus.zlibrary.core.constants	各种命名空间，其实就是各种url链接
org.geometerplus.zlibrary.core.encoding	用于读取encoding/Encondings.xml内的编码
org.geometerplus.zlibrary.core.filesystem	里面有各种File类，比如压缩File，资源的File，本地File等
org.geometerplus.zlibrary.core.filesystem.tar	Rar文件头，字节流封装类
org.geometerplus.zlibrary.core.html	html解析
org.geometerplus.zlibrary.core.image	图片加载，图片处理相关的包
org.geometerplus.zlibrary.core.language	编码检测包
org.geometerplus.zlibrary.core.library	有获取FBReader版本，调节屏幕亮度的包
org.geometerplus.zlibrary.core.network	Cookie数据库建立，网络请求，网络管理相关的包
org.geometerplus.zlibrary.core.options	FBReader自定义的数据类型，比如ZLBoolean3Option,ZLColorOption
org.geometerplus.zlibrary.core.resources	本地资源文件
org.geometerplus.zlibrary.core.sqliteconfig	配置数据库的建立
org.geometerplus.zlibrary.core.tree	树结构，ZLTree
org.geometerplus.zlibrary.core.util	各种工具类
org.geometerplus.zlibrary.core.view	自定义的View类，是阅读界面显示的最高抽象类，以及事件监听，处理
org.geometerplus.zlibrary.core.xml	解析XML
org.geometerplus.zlibrary.text.hyphenation	根据断字来判断是那种语言
org.geometerplus.zlibrary.text.model	一个抽象类包，有对文字的处理，以及对段落的处理，保存文字到缓存
org.geometerplus.zlibrary.text.view	文字被结构化，文字块，一行文字，文字段，一页，都在 此包下处理。以及显示的View
org.geometerplus.zlibrary.text.view.style	基本是文字装饰，高亮，链接的相关的装饰类。
org.geometerplus.zlibrary.ui.android.application	自定义的管理Window的应用
org.geometerplus.zlibrary.ui.android.image	图片获取和管理
org.geometerplus.zlibrary.ui.android.library	错误报告，程序的主Activity,Application在此包抽象化
org.geometerplus.zlibrary.ui.android.network	Cookie数据库操作
org.geometerplus.zlibrary.ui.android.util	颜色处理工具类
org.geometerplus.zlibrary.ui.android.view	各种翻页效果
org.vimgadgets.linebreak	断出一个“字”，假设“我是，啊！”“，断出来就是“我”“是”，“啊！”“。三个字。

下面就是C/C++部分简单的解析，ini文件夹下：



文件夹主要的结构就是如此，其他的比如Assets、Icons、Obsolete文件夹下，就由大家自己去了解。到这里，文件夹就基本搞定。

#### 四、重要文件分析

首先，编译C/C++的时候，需要编写NDK能够识别的Android.mk文件。它规定了整个编译的流程，还有需要调用的包，非常重要。

下面看几个比较重要的两部分：

```
1.
include $(CLEAR_VARS)
LOCAL_MODULE := DeflatingDecompressor
LOCAL_SRC_FILES := DeflatingDecompressor/DeflatingDecompressor.cpp
LOCAL_LDLIBS := -lz
```

上面的代码，注意最后的LOCAL\_LDLIBS就可以了。因为是解压缩功能，所以会调用ZLIB包，于是就加入"-lz"。

2.



```

include $(CLEAR_VARS)

EXPAT_DIR := expat-2.0.1

LOCAL_MODULE             := expat
LOCAL_SRC_FILES          := $(EXPAT_DIR)/lib/xmlparse.c $(EXPAT_DIR)/lib/xmlrole.c
LOCAL_CFLAGS             := -DHAVE_EXPAT_CONFIG_H
LOCAL_C_INCLUDES         := $(LOCAL_PATH)/$(EXPAT_DIR)
LOCAL_EXPORT_C_INCLUDES  := $(LOCAL_PATH)/$(EXPAT_DIR)/lib

include $(BUILD_STATIC_LIBRARY)

include $(CLEAR_VARS)

SHAREDIR_MACRO := nativeShare

LOCAL_MODULE := NativeFormats
LOCAL_LDLIBS := -lz
LOCAL_CFLAGS := -Wall -DBASEDIR=\"$(SHAREDIR_MACRO)\"
LOCAL_STATIC_LIBRARIES := expat

LOCAL_SRC_FILES := \
    NativeFormats/extensions.cpp \

```

用“include \$(CLEAR\_VARS)”做分割，前面有个编译出来expat，其实是一个解析XML的包；后面的NativeFormats的编译需要用到expat包，所以会先让expat编译，然后用LOCAL\_STATIC\_LIBRARIES:=expat来调用！

其次，AndroidManifest.xml文件也是很重要的。

## 1.Application

```
<application android:name="org.geometerplus.zlibrary.ui.android.library.ZLAndroidApplication"
```

从这里知道FBReader的Application是ZLAndroidApplication。

项目代码多，有时很多名字很相似，会误导我们，记住ZLAndroidApplication!

## 2.MainActivity (就是主Activity)

```

<activity android:name="org.geometerplus.android.fbreader.FBReader" ;
    <intent-filter>
        <action android:name="android.intent.action.MAIN" />
        <category android:name="android.intent.category.LAUNCHER" />
    </intent-filter>
</activity>

```

这个是以后跟踪程序的出发点，信息有用,FBReader，嘿嘿！

## 3.就是permission

```

<uses-permission android:name="android.permission.INTERNET" />
<uses-permission android:name="android.permission.WAKE_LOCK" />
<uses-permission android:name="android.permission.WRITE_EXTERNAL_STORAGE" />

```

上面有INTERNET权限，是网络连接，还有WRITE\_EXTERNAL\_STORAGE,这个就是写SDcard的权限了，猜猜会拿来干啥？

## 五、简单框架

做个简单的比喻，也许不太恰当，但是可能对一部分人有更好的理解。

FBReader是一个复杂的项目（对我刚入此行来说），代码量相当庞大。如何把它分解出几个局部，再来分析，肯定会对代码有一个比较好的方法。

- com.nil.util
- org.amse.zip
- org.geometerplus.android.fbreader
- org.geometerplus.android.fbreader.api
- org.geometerplus.android.fbreader.crash
- org.geometerplus.android.fbreader.image
- org.geometerplus.android.fbreader.library
- org.geometerplus.android.fbreader.network
- org.geometerplus.android.fbreader.preferences
- org.geometerplus.android.fbreader.tree
- org.geometerplus.android.util
- org.geometerplus.fbreader
- org.geometerplus.fbreader.bookmodel
- org.geometerplus.fbreader.fbreader
- org.geometerplus.fbreader.formats
- org.geometerplus.fbreader.formats.fb2
- org.geometerplus.fbreader.formats.html
- org.geometerplus.fbreader.formats.oeb
- org.geometerplus.fbreader.formats.pdb
- org.geometerplus.fbreader.formats.plucker
- org.geometerplus.fbreader.formats.util
- org.geometerplus.fbreader.formats.xhtml
- org.geometerplus.fbreader.library
- org.geometerplus.fbreader.network
- org.geometerplus.fbreader.network.atom
- org.geometerplus.fbreader.network.authentication
- org.geometerplus.fbreader.network.authentication.litres
- org.geometerplus.fbreader.network.opds
- org.geometerplus.fbreader.network.tree
- org.geometerplus.fbreader.network.urlInfo
- org.geometerplus.fbreader.tree
- org.geometerplus.zlibrary.core.application
- org.geometerplus.zlibrary.core.config
- org.geometerplus.zlibrary.core.constants
- org.geometerplus.zlibrary.core.encoding
- org.geometerplus.zlibrary.core.filesystem
- org.geometerplus.zlibrary.core.filesystem.tar
- org.geometerplus.zlibrary.core.html
- org.geometerplus.zlibrary.core.image
- org.geometerplus.zlibrary.core.language
- org.geometerplus.zlibrary.core.library
- org.geometerplus.zlibrary.core.network
- org.geometerplus.zlibrary.core.options
- org.geometerplus.zlibrary.core.resources
- org.geometerplus.zlibrary.core.sqliteconfig
- org.geometerplus.zlibrary.core.tree
- org.geometerplus.zlibrary.core.util
- org.geometerplus.zlibrary.core.view
- org.geometerplus.zlibrary.core.xml
- org.geometerplus.zlibrary.text.hyphenation
- org.geometerplus.zlibrary.text.model
- org.geometerplus.zlibrary.text.view
- org.geometerplus.zlibrary.text.view.style
- org.geometerplus.zlibrary.ui.android.application
- org.geometerplus.zlibrary.ui.android.image
- org.geometerplus.zlibrary.ui.android.library
- org.geometerplus.zlibrary.ui.android.network
- org.geometerplus.zlibrary.ui.android.util
- org.geometerplus.zlibrary.ui.android.view
- org.vimgadgets.linebreak

5

5.这里就是一些皮，手指，眼睛等，实现对外部世界的感应。其实就是FBReader对事件，对文字显示，对网络支持，对配置的处理。它是一个外部的表现，是人真正的外表！

4

4.这里才是血液之源，它是各个格式的解析，只有从这里，才有数据出来。它和3,2,1构成了整个框架的锥形。

6

6.网络部分，就好比人伸手去吃东西一样。它类似于食物，能够让我们得到能源。

1

1.其中的类基本是抽象类abstract,或者最终类final. 抽象类中，留有很多抽象方法给子类实现。正如作者取名的core一样，却是是整个程序的核心。但是它是一个骨架，没有肌肉。

3

3.它基本也是抽象类，但是它留有对文字处理的方法，可把他看成血液，它和2构成了五官。

2

2.如果说1是一个骨架，那么2就是肉。这里是一个UI相关的包，但是它还是一个没有血液，就好像画画给你一张纸一样。它要到3,4中获取数据。

嗯。上面的文字描述，可能没有尽析FBReader框架的精妙，但是早前已说我功底尚浅，各位童鞋就谅解下。

此篇文章就分析到这里！

对FBReader感兴趣的童鞋请加入：

QQ群：218971429，233932132

或者访问我们“六阅雪”网站：

<http://sixreader.com>

\_\_END\_\_

(转载请说明出处----来自六阅雪)