|  |
| --- |
| **FxLib设计文档** |
| 附测试数据 |
|  |
| 本文对FxLib的框架以及设计进行说明 |
|  |
| **倪杰** |
| **2010-12-28** |
|  |

**[倪杰同学个人所有]**

目录

[1 服务器构架概述 3](#_Toc281333327)

[2 GameServer设计 4](#_Toc281333328)

[2.1 模块说明 4](#_Toc281333329)

[2.2 服务器对象继承树（对象模型） 5](#_Toc281333330)

[2.2.1 UML图 5](#_Toc281333331)

[2.2.2 说明 5](#_Toc281333332)

[3 底层库设计 7](#_Toc281333333)

[3.1 windows版网络库线程模型 7](#_Toc281333334)

[3.2 linux版网络库线程模型 8](#_Toc281333335)

[3.3 压力测试结果 9](#_Toc281333336)

# 服务器构架概述

这里列举的是一般游戏的服务器架构，特别是针对MMORPG游戏的框架设计。针对不同的游戏类型，由于偏重点不一样，要做适当的修改。比如ACT游戏，这里的GateServer对于战斗逻辑服务器可能会带来一些不必须要的延迟。对于一般的小游戏，原则上就是建议使用尽量少的服务器进程，以便更好的开发和维护，减少维护工作量，可以更大程度的减少出错率。

服务器种类及说明：

* **CenterServer**：用于保证玩家的唯一性等；
* **DBServer**：数据缓冲服务器，为了减少对数据库的读写次数，提高效率；
* **LoginServer**：认证玩家信息，调用运营环境的登录模块，连接运营环境的AccountServer；
* **RelationServer**：关系服务器，处理组队、公会、好友等社会关系；
* **GameServer**：处理主要的游戏逻辑；
* **GateServer**：网关服务器，处理与客户端的网络通信。

结构图如下所示：



# FxLib库设计

## FxLib概述

FxLib包括一下功能组件：

FxNet 大型网络服务器组件

FxDB MySql数据库接口封装

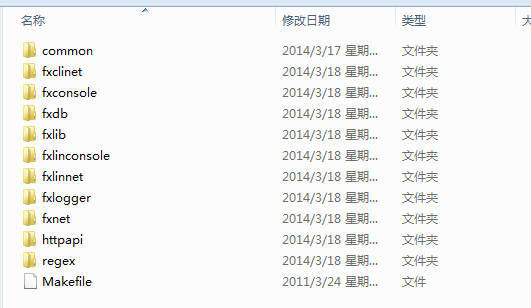
FxConsole 服务器Console管理库

FxLogger 服务器日志组件

HttpApi 高性能HTTP请求转发接口封装

Regex 正则表达式接口封装、

Fxclinet 游戏客户端网络封装



同时也制作、收集和优化一些服务器方面的工具

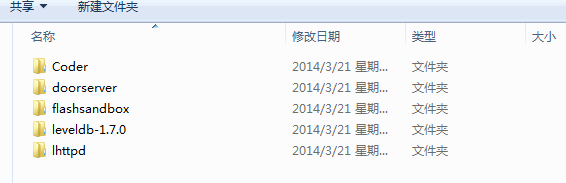
Doorserver 包含了服务器后台管理工具，以及一个管理客户端。

Flashsandbox 自制flash安全沙箱服务器。

Leveldb google开源的一个超高性能的K-V数据库，优化

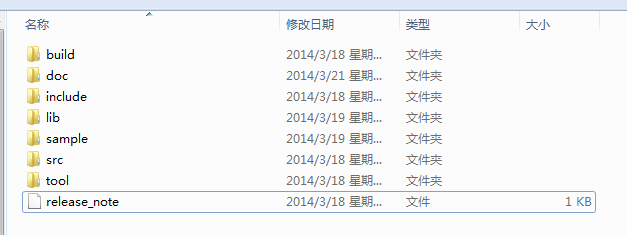
Lhttpd 自制的httpserver，后台脚本使用lua（还在开发中）。

Coder 协议自动生成工具

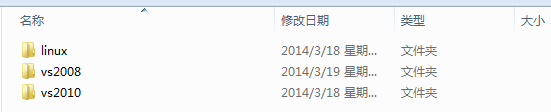


## FxLib 目录结构

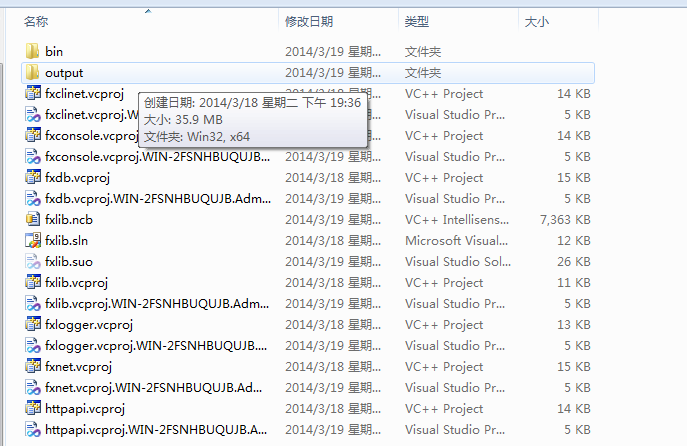
截图如下：



* Build目录为编译工程目录

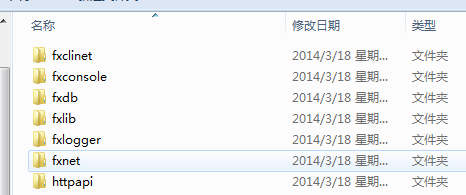
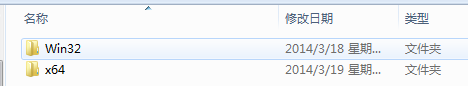


分成3个版本，分别是vs2008，vs2010，和linux版本编译目录，编译完后会相应生成一个版本的库文件，例如：



Bin里面是生成的库文件

Output目录是编译的中间文件



* Doc为设计文档目录，包括此文档等一些服务器模块的文档
* Include 为FxLib组件接口头文件目录
* Lib目录为FxLib组件导出库文件目录
* Sample目录为FxLib组件例子Demo和测试用例存放目录
* Src为FxLib源码目录 -- 不发布
* Tool为本人收集、优化和自制的一些服务器小工具及其源码存放的目录。

## FxNet网络库组件

FxNet是大型网络库组件，也是FxLib中最最重要的服务器组件。他支持windows和linux两个版本的服务器，同时暴露的接口是完全一致的。所以逻辑开发者在开发和调试时只要在windows下进行即可，到最后出测试、发行版本的时候编译出linux版本，即可在linux服务器上运行，这样可以大量节省由于linux开发不熟，或者工具不齐全引起的效率低下，也可以直接使用windows版服务器，在windows2008服务器上测试，性能丝毫不差。

其暴露给上层开发者的接口也很简单：



### windows版网络库（FxNet）线程模型

下图为网络库线程模型：



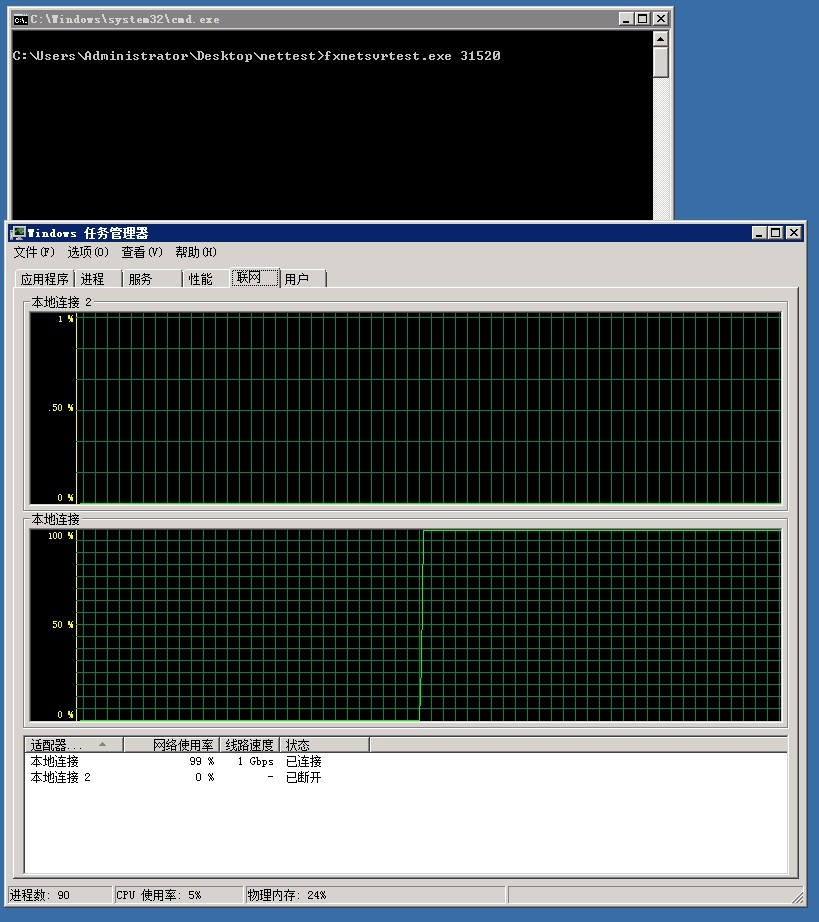
### linux版网络库（FxLinNet）线程模型



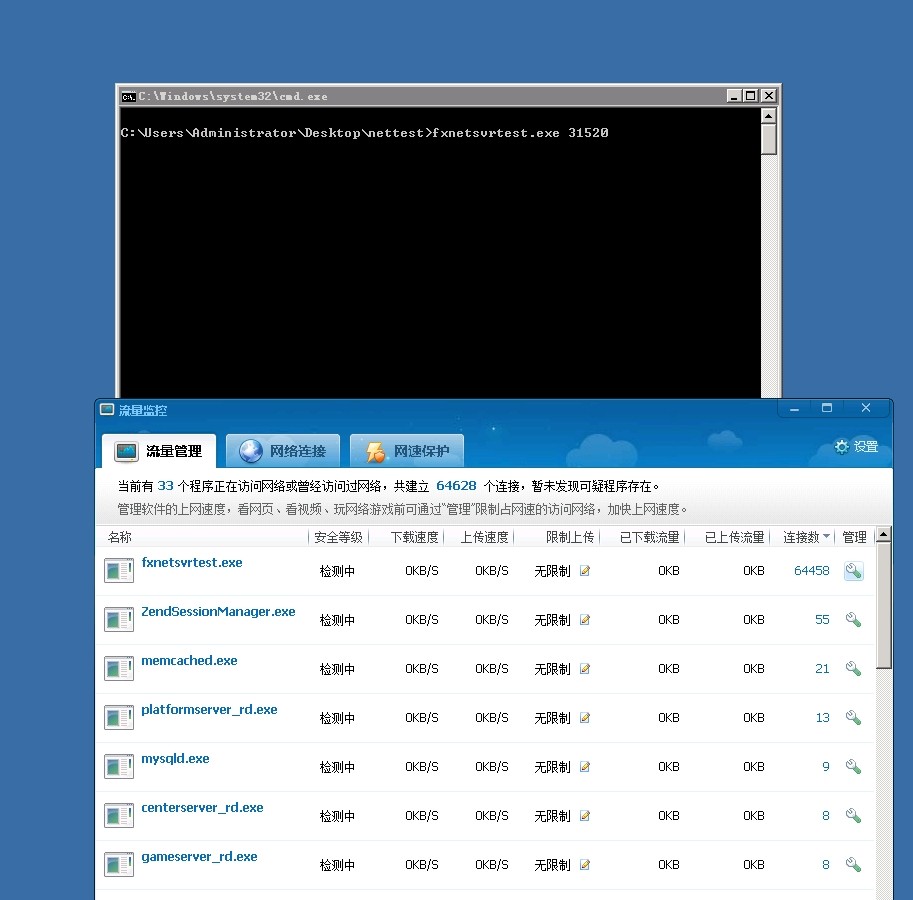
### 网络库压力测试结果

windows版本网络库，15000多个连接，1000Bytes的数据包连续发送情况下的CPU及网络状况截图：

* CPU 网络压力99%近100%时，cpu占用5%，内存占用24%



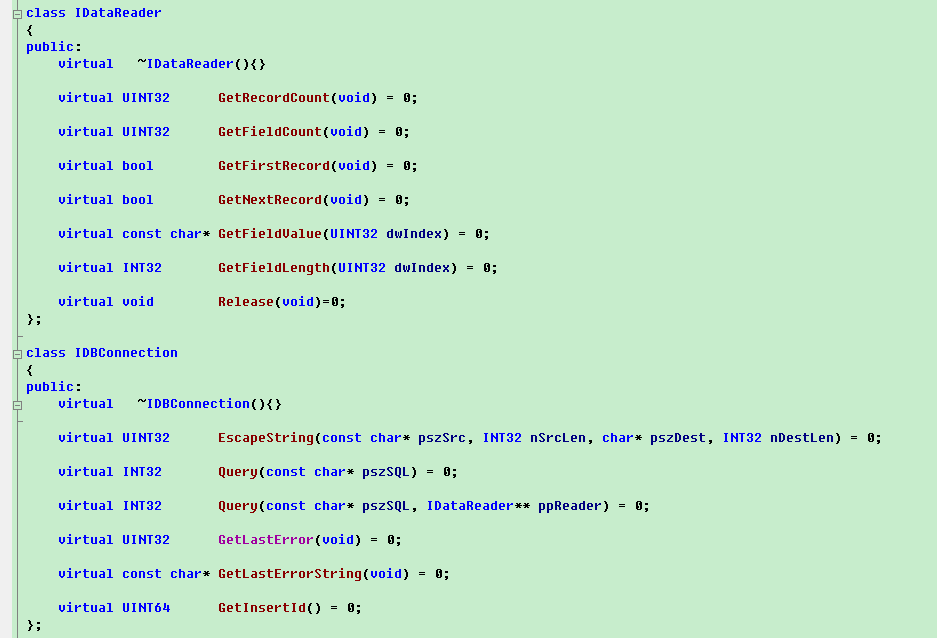
* 连接数测试 在windows2008上直接可以达到60000多个连接管理。（没有再网上测试，也没有在60000个连接时做压力测试，因为360网络检测模块在这个时候会占用服务器大量负载，没法有效测试）



## FxDB组件

FxDB组件是对MySql查询接口的一次封装，主要是使其查询过程多线程，同时上层逻辑单线程，这样开发者就不需要考虑查询连接死锁和主线程性能及互斥问题，专注于逻辑开发。

接口如下：



线程模型如下

## 

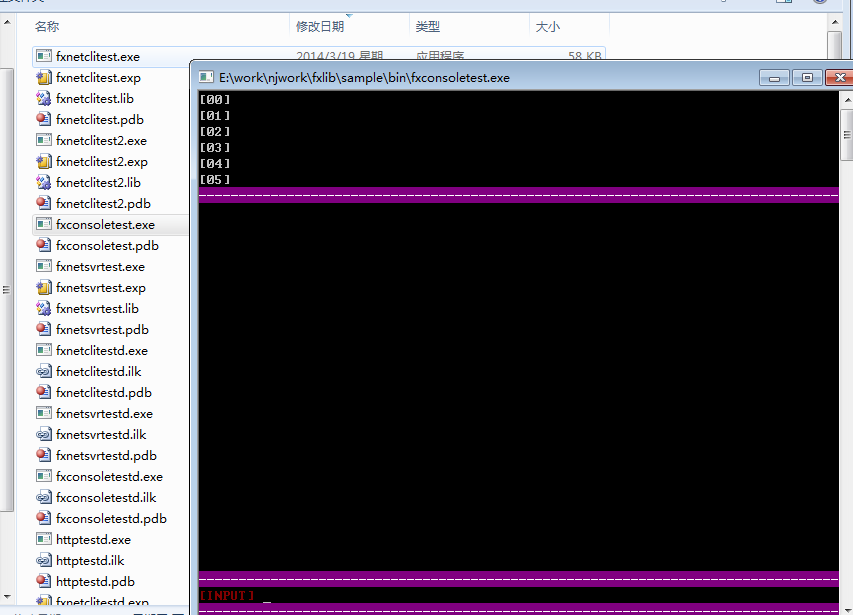
## FxConsole 服务器Console管理库

该组件主要用于服务器Console控制台及其命令行管理。

接口如下：



运行Demo如下：

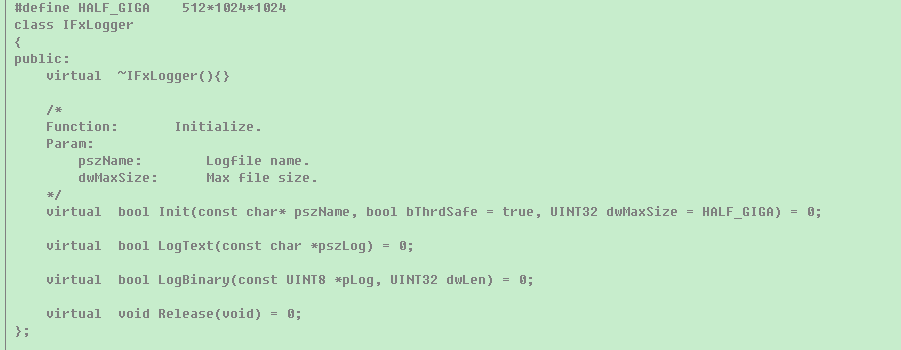


上方为日志或者服务器打印消息等，下方INPUT可以出入自定义的各种指令。

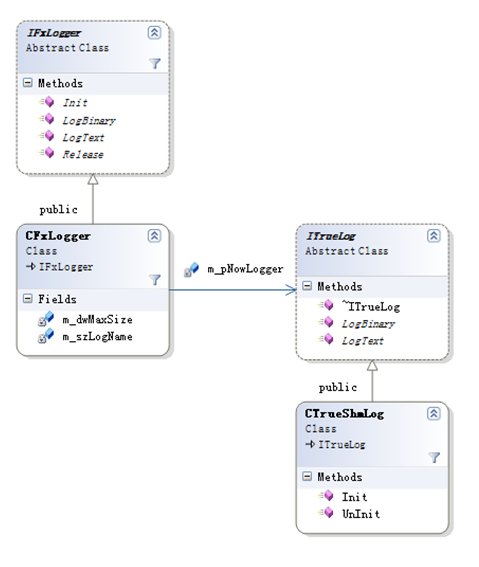
## FxLogger 服务器日志组件

FxLogger为服务器日志组件。其特点是使用虚拟内存方式，即使服务器进程异常崩溃，这样调用过日志组件记录的日志都不会丢失，系统在进程注销的时候会自动把内存中的数据保存到硬盘上。这样既提高性能，又保证日志安全。对于查找崩溃时的日志有很大的好处。

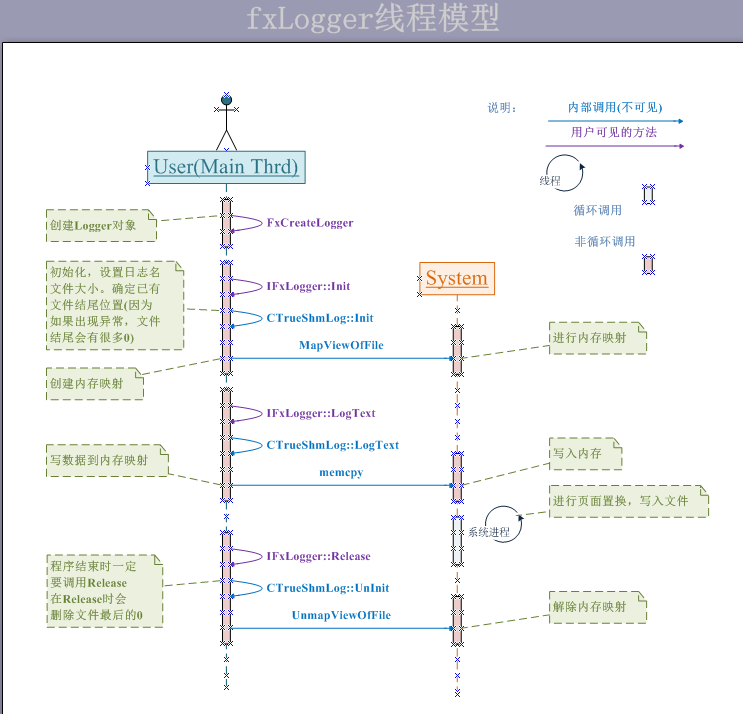
接口如下：



其类图如下:



其线程模型如下：



## HttpApi 高性能HTTP请求转发组件

HttpApi主要是对curl的二次封装。和FxDB一样，主要是让其主线程和工作线程（Http慢线程）分开，并在底层管理好，上层逻辑主线程不需要考虑耗性能的http请求过程，只要发送请求和处理结构即可。其工作原理及类图这里就省略了。其结构很简单，可参考fxdb。

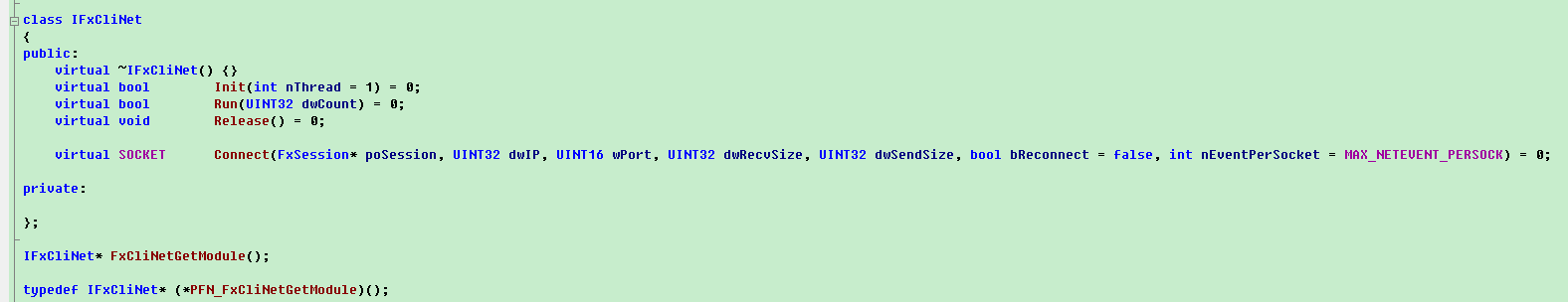
接口如下：



## Fxclinet 游戏客户端网络组件

客户端网络通信组件。使其接口和FxNet基本上一致。

接口如下：



## Regex 正则表达式接口封装

对PCRE的二次封装。