1.服务器框架结构

1.1基本结构

服务器的主类继承GameService这个类，然后通过main函数启动就可以；

服务器的功能模块增加adapter来实现，从BaseAdapter继承；

每一个adapter可以有n个儿子节点；

ServiceID,AdapterID,ObjID可以唯一确定一个obj，服务器内部消息全部以这个格式来传递；

Service启动以后，首先会向Service mgr注册，所以每一个Service的ID必须是唯一的；

1.2服务器分类

1.2.1 Game Server（基本不需要改动）

整个服务器群的大脑，也是唯一的，第一个启动，Service mgr也在这个上面；

iPad版的管理工具也是把命令发到这里，由他负责响应或者转发；

1.2.2 DB Server

数据库服务器

功能：读写数据库，数据库缓存

可以根据负载，通过配置增加多个DB Server分布式处理请求，目前为止看，一个足够

DB Server共有3种Adapter，如下

DBLoginMgr（基本不用改动）

负责登录时候的信息获取，

DBLoginHandler是他的儿子节点的类，login信息获取具体实现的类

DBUserMgr

查询角色和创建角色会用到

DBUserHandler是他的儿子节点的类，根据UserID，查询和创建角色具体实现

DBHomeMgr

管理每一个玩家角色的所有数据的读取和更新

DBHomeHandler是他的儿子节点的类，根据CharacterID，读取和更新玩家角色信息的具体实现

1.2.3 User Info Server

玩家角色全局信息管理服务器，只有一个，是唯一的；

把当前玩家的信息保存在内存中，，查找陌生人，好友具体信息，都是通过他来获得，也就是说获取别的玩家信息不会从数据库获得，而是从User Info Server获得；

PVP的排行榜和匹配池也全部由他来管理；

1.2.4 PhpProxy

作为php页面和内网服务器通信的一个代理，把php的json请求转成内网格式的消息转发，把内网格式的消息转成json转发给php；

目前为止只有内购的购买协议使用了他，包括ios和电信的，同一条协议；

1.2.5 Login Server（基本不需要改动）

登录服务器，只有一个

负责响应玩家的登录请求，返回合适的gateway ip地址和端口

1.2.6 Gateway

可以有多个，根据负载，通过配置增加多个Gateway

负责保证客户端合法连接

负责序列化和反序列化客户端和服务器之间传递的消息

1.2.7 Map Server

游戏的主逻辑服务器，可以根据负载，通过配置增加多个Map Server

Map类的实例负责一个玩家角色所有的游戏逻辑

1.2.8 DataUpdateTool

就是一个工具，负责版本更新以后的数据库数据转换

1.3 服务器启动顺序

a. Game Server

b. DB Server

c. User Info Server

d. Map Server

e. PhpProxy

f. Gateway

g.Login Server

1.3 添加新协议

协议添加在ControlProtocol.h中；

如果需要更客户端通信的，客户端请求继承结构MsgBaseRequest，返回给客户端的应答继承结构MsgBaseResponse

客户端用到的协议需要在Gateway中序列化和反序列化；

客户端的请求在GatewaySession类的函数virtual int parse\_into\_intranet\_msg(MessageInfo &msgInfo,char \*buff, int size)中处理,反序列化成内网的协议;

给客户端的应答在GatewaySession类的函数virtual ACE\_Message\_Block \*parse\_into\_internet\_msg(MessageInfo &msgInfo)中处理，序列化成客户端用的字节流；

1.4 添加新功能

1.4.1 客户端到服务器流程

客户端发送请求到gateway；

Gateway反序列化成内网协议，发送给Map Server;

Map Server中的Map处理，返回应答给Gateway，如果有数据更新，向DB Server发送更新请求(add\_db\_request函数)；

Gateway序列化成字节流返回给客户端

1.5 服务器配置和添加新服务器

服务器配置文件service\_config.xml

1.5.1 <multicast>

组播地址，现在不起作用

1.5.2 <database>

<name>xiyou\_db</name> 数据库名

<host>tcp://192.168.1.43:3306</host> 数据库地址

<user>root</user> 数据库用户名

<password>123456</password> 数据库密码

1.5.3 <service\_define\_list> 服务器定义列表

<service\_define>

<name>Game Server</name> 服务器定义名称

<id>1</id> 服务器定义ID

<is\_serviceMgr>1</is\_serviceMgr> ServiceMgr的标志位

</service\_define>

1.5.4 <service\_instance\_list> 服务器实例列表

<service\_instance>

<name>Game Server</name> 实例名字

<type>Game Server</type> 服务器定义名字

<lan> 内网地址

<ip>192.168.1.43</ip> ip地址

<port>1005</port> 端口

</lan>

<wlan> 外网地址

<ip>192.168.1.43</ip>

<port>1100</port>

</wlan>

<public> 公开发布地址

<ip>192.168.1.43</ip>

<port>1100</port>

</public>

<id>1</id>

</service\_instance>

2.PHP代码

2.1 文件列表

create\_account.php

创建新的帐号，在快速进入游戏的时候使用

DBInstance.php

php对数据库操作的代码都在这里

deliver\_sms\_i.php

电信短信充值以后，发放元宝的具体实现

deliver\_sms.php

电信短信充值以后，电信的回调php，这个php然后再调用deliver\_sms\_i.php

deliver.php

ios充值以后，发放元宝的具体实现

env.php

php中用到的环境变量，主要是一些ip地址和ios充值验证地址

login.php

登陆用php，获取user\_id和mac

lxy\_gonggao.html

公告

order.php

电信充值生成订单

register.php

注册帐号或者绑定帐号

server\_ip\_list.php

根据版本，返回对应的服务器ip

ServerRequest.php

php和内网服务器通信的类，用json做为消息传递的格式

sms\_order\_status.php

查询电信充值订单的状态

utils.php

一些工具函数的实现

3.维护必须要做的事

修改server\_ip\_list.php，使当前版本的客户端登陆显示维护

用ipad工具关闭服务器

备份服务器代码

备份数据库数据

更新服务器代码（可选）

更新公告（可选）

更新php（可选）

运行数据库数据转换工具（可选）

开启服务器

新版本测试（可选）

修改server\_ip\_list.php，取消维护状态