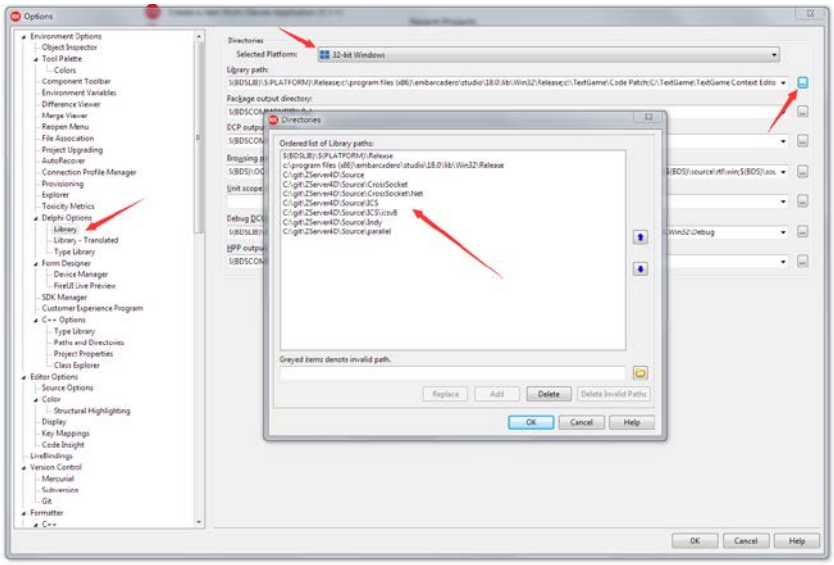
云服务器框架 2.0

目录

编译：

编译 Zserver4D 请将系统目录设置指向到 Source，并且确保指向了 Indy,CrossSocket,ICS 等等组件目录

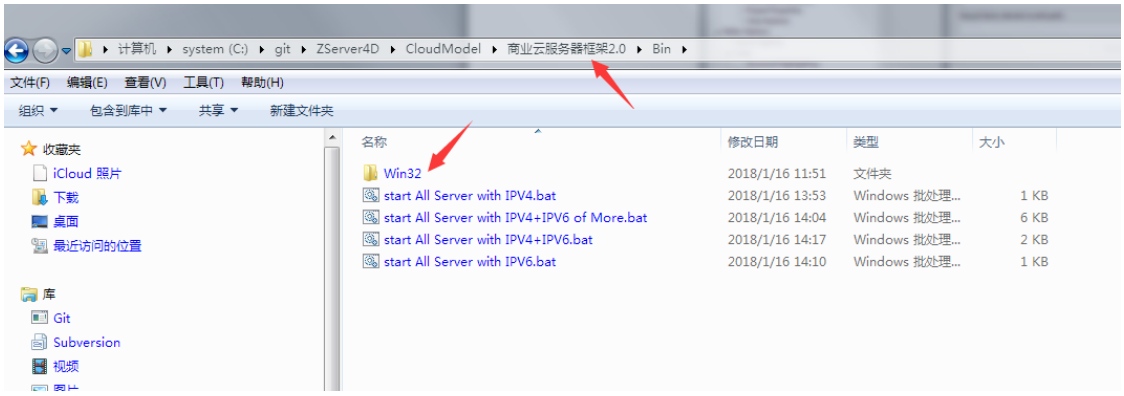


编译注意事项

Logic(逻辑服务器)，有引用一个叫 LogicFile.OX 的包文件，此文件使用 FilePackageWithZDB 工具进行的打包，编译它时尽量使用 Build，如果编译器发生了 Internal Error，请关闭 IDE，重启一次 Delphi，再次 Build

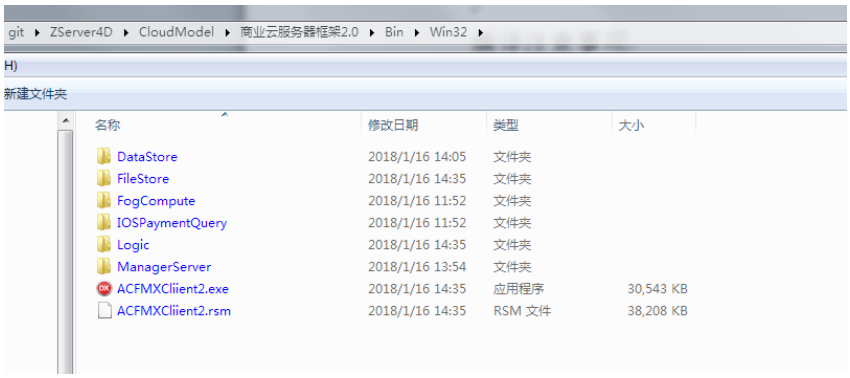
编译后的 EXE 在哪里？

所有服务器和客户端编译完成后，所有的 EXE 输出都会在 Bin 目录中，在 Bin 目录中的启动脚本，都是面向 Win32 平台的，如果你编译的是 x64 平台，那么请自己修改这些启动脚本



编译产生的文件说明

DataStore：数据库服务器
FileStore：文件存储服务器
FogCompute：雾计算服务器
Logic：业务服务器
ManagerServer：管理服务器
IOPaymentQuery：针对 IOS 平台的支付验单服务器
ACFMXClient2：云服务器的客户端



启动服务器的方法

编译成功后，在默认情况下，请使用 start All Server with IPV4+IPV6.bat 脚本来启动服务器，不出问题 5 秒内会全部启动完成

调试服务器的方法

首先，在编译时，需要先打开远程调试符号，然后用脚本启动一批服务器，
关键：这时候，用 **ATTACH TO PROCESS...**方式切入服务器调试
正常情况下，IDE 不需要多开，通过查看 Log 信息即可判断出服务器的运行状态，代码问题，直接 Attach to Process 即可，修改完代码后，重新拿脚本开出一批
服务器右上方的 STATUS 开关都是调试服务器用，发行服务器，此开关必须关闭，否则并发性能会下降 10 倍以上

开多台 Logic 业务服务器的注意事项

因为业务服务器使用的本地化的双通道认证框架，在本地开多台注意下列问题

1， 确保每台服务器的端口号是唯一的
2， 确保每台服务器都在一个独立的目录中，在同一个目录中，开多台业务服务器会出错，因为业务服务器使用的本地化的双通道认证框架，所有的用户都存在 UserDB，会发生冲突
3， 业务服务器一共有 4 个侦听端口需要定义

云服务器框架说明

- 云服务器的框架很大，请多读代码，遇上无法理解的问题时，可以到 QQ 群给我留言
- 云服务器框架支持百万级并发服务
- 云服务器框架支持 IPV6+IPV4，可以安全过审苹果
- 云服务器框架支持内网穿透
- ManagerServer 可以一机多开，支持无中心化 ManagerServer
- ManagerServer 内部逻辑均在 CommunicationFrameworkDoubleTunnelIO_ServMan.pas 中实现
- FileStore 文件服务器可以多开，实现分布式负载
- DataStore 数据服务器只能开一台，假如做分布式，请在 DataStore 内部自行实现分流
- ManagerServer 内部对手机平台提供了最小负载查询，和全局服务器查询两种方法
- 在 Logic 的业务服务器中，包含了两套服务，分别是业务服务本身和支付服务，但是支付服务没有任何业务代码
- 针对数据库 DataStore 的查看，云服务器框架提供了 DataStoreViewClient.exe（包含源码），可以算作是查询后台数据库的 Demo 和开发范式
- 业务服务器使用 LogicFile.OX 是避免过多零碎的业务文件对管理服务器造成不便
- 业务服务器在用户登录时，会从 FileStore 下载用户的副本并且验证，直到用户离线时，业务服务器会把用户登录副本重新上传一份到 FileStore
- 业务服务器会记录用户的所有登录信息到 DataStore 服务器中
- 业务服务器的所有 Status 都会记录到 DataStore 服务器中
- 雾计算服务器的所有 Status 和计算公式都会记录到 DataStore 服务器中
- FileStore 服务器的所有 Status 都会记录到 DataStore 服务器中

ACFMXClient2.0 框架说明

- 客户端支持自动化的断线重连
- 客户端支持自动化 UI 缩放（1080p）
- 客户端支持自动化 IPV6 和 IPV4 的试探性链接
- 客户端支持零阻塞
- 客户端支持对 Logic 服务器中的 LogicFile.OX 内容快速检索和下载
- 客户端支持运行于 Android+IOS 手机平台
- 客户端支持消耗型项目内购 支持库：IAP_IOSImp.pas
- 客户端的 Form 结构和全局定义可以作为开发范式参考
- 客户端内置了安卓和苹果的反屏幕空闲功能
- 客户端的 Progress 都是在内置的线程中运行，假如担心耗电，可自行加入 Sleep 等待

二次开发方法

- 先将框架复制到你的工程目录
- 修改 ServerManTypeDefine.inc，将你的服务器种类，写进 TserverType 的定义中
- 所有的服务器的默认端口都在 CommonServiceDefine.pas 中配置
- 通常情况下，我们做扩展的业务开发，直接针对 Logic 做修改就行了，当我们实际运行业务时，将 Logic 部署到不同的实体服务器去即可完成大规模分布式负载
- FOGCompute 雾计算服务器是密集型服务器，当你服务器配备了多核 cpu 和海量内存时，可以一机多开，也可以用于分布式部署