

InIOCP（小旋风服务套件）

# 组 件 使 用 说 明

（版本：2.5.30.1221）

适用版本：Delphi 7、Delphi 2007、  
Delphi XE、Delphi XE5-XE10

作 者：高凉新农，QQ 群：365531817

下载途径：csdn.net，2ccc.com 的 ftp 资源

github: <https://github.com/cnwneumann/InIOCP>

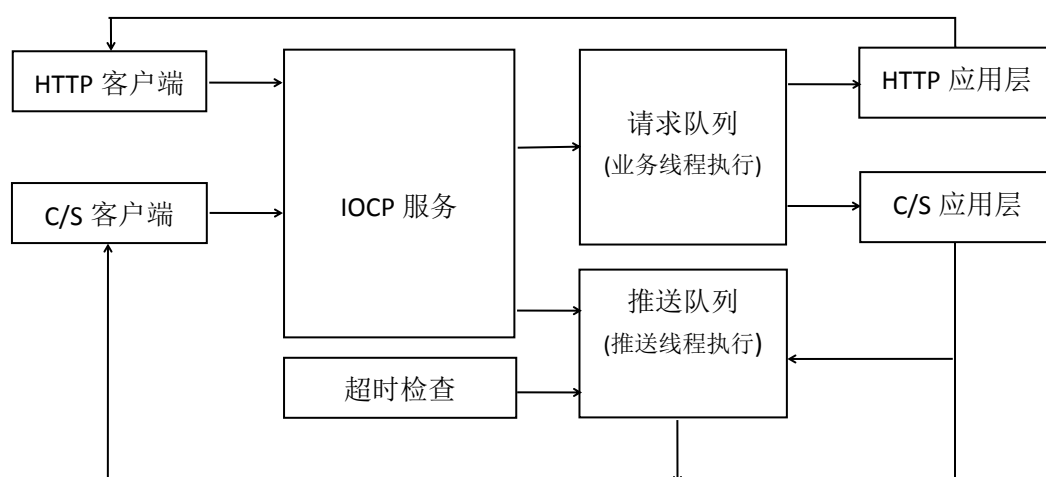
在广大热心网友的支持下，InIOCP 小旋风服务套件稳定版已经顺利发布，在此再次感谢大家！

InIOCP 发布这么久，一直没写过使用说明，现在抽点时间简单介绍一下，写多少就发布多少吧，也欢迎各位网友反馈意见。

## 一、系统架构

下面的架构图很不专业，能理解就行，将就一下，箭头方向是请求或反馈数据的流向（严格来说反馈数据也经 IOCP 服务）：

图 1、InIOCP 架构简图



## 二、组件概况

（一）TInIOCPServer 是 IOCP 服务的核心，提供全部服务的最重要组件；TInConnection 是连接到服务器的客户端组件。

（二）目前，InIOCP 套件按业务（服务）类型分为 6 大类，每类又分为服务器端和客户端两部分，两端的组件有对应关系，请看下表：

表 1、InIOCP 组件分类及用途

编号	服务类型	服务端组件	客户端组件	用途说明
1	认证	TInClientManager	TInCertifyClient	用于客户端登录、登出、注册、查询和删除等操作。
2	消息	TInMessageManager	TInMessageClient	用于发送、推送、广播文本消息。
3	文件	TInFileManager	TInFileClient	用于上传、下载文件（支持断点续传）。
4	数据库	TInDatabaseManager	1、TInDBConnection 2、TInDBQueryClient 3、TInDBSQLClient	1、查询、设置数据库连接 2、执行 SELECT-SQL 3、执行非 SELECT 类型 SQL
5	自定义	TInCustomManager TInRemoteFunctionGroup	1、TInCustomClient 2、TInFunctionClient	1、发送自定义消息 2、执行远程函数
6	HTTP	TInHttpDataProvider	暂无	提供 HTTP 基本服务

注：表中 1-5 项均为 Client/Server 模式。

C/S 模式下，InIOCP 将客户端的请求操作进行分类，服务端根据客户端提交的请求操作而提供相应的服务，用户也可以结合需求对操作类型进行扩展，请参考 iocp\_base 单元的 TActionType 类型定义。

### 三、TInIOCPServer 组件

TInIOCPServer 是整个 IOCP 服务的核心，它提供上述几大服务，当然，为了实现这些服务，必须为其配置相应的管理组件（见表 1），下表是 TInIOCPServer 组件的主要属性：

表 2、TInIOCPServer 组件的主要属性

编号	属性名称	用途说明	代码使用处
1	Active	开启、停止服务。	

2	HttpDataProvider	关联提供 <b>HTTP</b> 服务的 THttpDataProvider 组件，为空时无此服务。	
3	ClientManager	关联提供 <b>认证</b> 服务的 TInClientManager 组件，为空时无此服务。	
4	CustomManager	关联提供 <b>自定义</b> 服务的 TInCustomManager 组件，为空时无此服务。	
5	DatabaseManager	关联提供 <b>数据库</b> 服务的 TInDatabaseManager 组件，为空时无此服务。	
6	FileManager	关联提供 <b>文件</b> 服务的 TInFileManager 组件，为空时无此服务。	
7	MessageManager	关联提供 <b>消息</b> 服务的 TInMessageManager 组件，为空时无此服务。	
8	ServerAddr	服务地址（如本地 IP 或 localhost）。	
9	ServerPort	服务端口（默认为 12302）。	
10	BusyRefuseService	繁忙时 <b>拒绝</b> 服务。	TInIOCPServer.AcceptClient
11	ClientPoolSize	客户端池初始大小，池满时如果 BusyRefuseService 为真则 <b>拒绝连接</b> 。	TInIOCPServer.AcceptClient
12	MaxPushCount	推送消息每秒最大值，超过时 <b>拒绝推送</b> 。	
13	MaxQueueCount	任务队列最大堆积数，超过时 <b>拒绝连接</b> 。	TInIOCPServer.AcceptClient
14	PreventAttack	IP 级别 <b>防攻击</b> 。	TInIOCPServer.AcceptClient
15	TimeOut	超时间隔（毫秒，为 0 时使用心跳，不用系统的超时检查）。	TTimeoutThread.ExecuteWork
16	BusinessThreadCount	<b>业务线程</b> 数（依硬件而定，一般几个即可）。	TBusiWorkManager.Create
17	PushThreadCount	<b>推送线程</b> 数（依硬件而定，一般几个即可）。	TPushMsgManager.Create
18	WorkThreadCount	IOCP <b>工作线程</b> 数（标准是按 CPU 内核数定，可灵活设定，一般几个）。	TWorkThreadPool.Create

如果 TInIOCPServer 组件不关联任何管理组件，则只提供数据流服务，此时数据传输触发 OnDataReceive、OnDataSend 两个的事件，后者不能操作数据（见流服务例子）。

下面看 TInIOCPServer 组件的事件：

表 3、TInIOCPServer 组件的事件

编号	事件名称	用途说明	代码使用处
1	AfterClose	停止后触发的事件。	TInIOCPServer.InternalClose
2	AfterOpen	成功启动后触发的事件。	TInIOCPServer.OpenDetails
3	BeforeClose	停止之前触发的事件。	TInIOCPServer.InternalClose
4	BeforeOpen	启动之前触发的事件。	TInIOCPServer.OpenDetails
5	OnDataReceive	收到数据时触发的事件（此时未进入业务层，不要操作数据）。	TWorkThread.ExecIOEvent
6	OnDataSend	数据发出后触发的事件。	TWorkThread.ExecIOEvent
7	OnError	异常事件。	TWorkThread.ExecuteWork

## 四、TInConnection 组件

TInConnection 是用于连接服务器的客户端组件，其他客户端组件与其关联，它的一个实例就是一个连接。

下表是 TInConnection 组件的属性：

表 4、TInConnection 组件的事件

编号	属性名称	用途说明	代码使用处
1	Active	开启、停止连接。	
2	AutoConnect	自动连接（可配合 ReuseSessionId 免再次登录）。	TInBaseClient.CheckState
3	LocalPath	接收数据的本地存放路径。	TRecvThread.Create
4	MaxChunkSize	断点续传时每次传输的最大长度（字节）。	TClientParams.ReadDownloadInf TClientParams.ReadUploadInf
5	ReuseSessionId	关闭时保留本次的登录凭证，再次连接后免登录即可发送消息。	TInConnection.InternalClose TInCertifyClient.HandleMsgHead
6	ServerAddr	服务器地址（IP 或域名）。	TInConnection.InternalOpen
7	ServerPort	服务器端口（默认为 12302）。	TInConnection.InternalOpen

在这里也要对客户组件的事件作个总说明，客户端事件分主动和被动两类：

**1、主动事件：**是客户端提交请求后，服务端返回数据触发的事件，全部客户端组件都有主动事件 OnReturnResult；

**2、被动事件：**是服务端向客户端发送数据，客户端收到后触发的事件，只有 TInConnection 有这种事件，就是 OnReceiveMsg，收到的这种消息就是**推送消息**，即其他客户端或服务端传输过来的消息，这种消息有很大的不确定性，既不知道何时收到，也不知道会收到多少。

几乎全部客户端组件的事件都带 TResultParams 类型的参数 Result，它对应服务端的返回结果 TReturnResult 类型的对象 Result。

下表是 TInConnection 组件的事件：

表 5、TInConnection 组件的事件

编号	事件名称	用途说明	代码使用处
1	AfterConnect	连接成功后触发的事件。	TInConnection. InternalOpen
2	AfterDisconnect	断开后触发的事件。	TInConnection. InternalClose
3	BeforeConnect	连接前触发的事件。	TInConnection. InternalOpen
4	BeforeDisconnect	断开前触发的事件。	TInConnection. InternalClose
5	OnAddWork	投放消息时触发的事件（不要在此改消息内容）。	TSendThread. AddWork
6	OnDataReceive	收到数据时触发的事件。	TInConnection. ReceiveProgress
7	OnDataSend	发出数据时触发的事件。	TInConnection. SendProgress
8	OnError	异常事件。	TInConnection. DoServerError TInConnection. DoThreadFatalError 等
9	OnReceiveMsg	被动接收到消息时触发的事件。	各客户端组件的 HandlePushedMsg

10	OnReturnResult	发出请求后服务端返回数据触发的事件。	各客户端组件的 HandleFeedback
----	----------------	--------------------	------------------------

## 五、认证服务

在这里先作个总说明, Client/Server 模式服务端的全部管理组件中, 除了 TInDatabaseManager 没有事件 (不公开) 外, 其他几个管理组件的事件基本都带有三个参数, 它们是 (详见单元 iocp\_managers、iocp\_socket) :

**1、Sender:** 是类 TBusiWorker 的对象, 它有两个重要属性 **DataModule** 和 **DataModules**, 前者是注册到管理组件 TInDatabaseManager 中的数模的第一个实例, 后者是全部实例的数组, 可以用它们操作数据库, 如:

```
TBusiWorker(Sender).DataModule.ExecQuery(Params:TReceiveParams;Result:TReturnResult);
```

此外, TBusiWorker(Sender).GlobalLock 是一个全局锁, 可根据需要使用。

**2、Params:** 是类 TReceiveParams 的实例, 其内容是客户端传递过来的消息, 可以用预设属性或 As 系列属性取出字段值, 如 Action、UserName、AsString[ '???' ] 等。

**3、Result:** 是类 TReturnResult 的实例, 用于反馈消息给客户端。一般地, 用 Result.ActRresult 返回操作结果, 用预设属性或 As 系列属性返回其他数据, 如 UserName、AsInteger[ '???' ] 等。

服务端和客户端均可用 AsStream 或 AsDocument 传输小文件, 如果文件比较大 (几兆或更大), 建议用 LoadFromFile, 此时文件当作附件发送, 不占用额外的内存开销。

切记，Params 和 Result 都带属性 Socket，它是对应服务端类 TIOCPSocket 的对象。

在此重复一下，C/S 模式服务端管理组件用于执行相应的操作 TActionType（见单元 iocp\_base），InIOCP 把开发常用的操作分类，客户端发出的操作，服务端会根据其分类投放给相应的管理组件中，后者执行相应的事件。

下表是服务端认证管理组件 TInClientManager 的事件：

表 6、TInClientManager 组件的事件

编号	事件名称	对应客户端操作	对应客户端组件方法
1	OnDelete	arUserDelete	TInCertifyClient.Delete
2	OnModify	atUserModify	TInCertifyClient.Modify
3	OnLogin	atUserLogin	TInCertifyClient.Login
4	OnLogout	atUserLogout	TInCertifyClient.Logout
5	OnRegister	atUserRegister	TInCertifyClient.Register
6	OnQueryState	atUserState	TInCertifyClient.GetUserState
7	不公开	atUserQuery	TInCertifyClient.QueryClients

下表为认证客户端组件的属性和事件：

表 7、TInCertifyClient 组件的属性和事件

编号	分类	名称	说明
1	属性	Connection	关联的 TInConnection。
2		Group	用户分组，未用。
3		Password	登录密码/口令。
4		UserName	登录用户名称。
5	事件	OnCertify	登录、登出时服务端返回结果触发事件。
6		OnListClients	查询在线用户时，服务端返回结果触发的事件。
7		OnReturnResult	发出各种认证类请求后，服务端返回结果触发的事件。



触类旁通，其他几大服务的服务端、客户端组件的介绍会简单一点，主要以列表方式解释（详细见 iocp\_base、iocp\_manager 和 iocp\_client 等单元）。

## 六、消息服务

表 8、TInMessageManager 组件的事件

编号	事件名称	事件说明	对应客户端操作	对应客户端组件方法
1	OnBroadcast	广播（发消息给全部客户端）。	atTextBroadcast	TInMessageClient.Broadcast
2	OnGet	取离线消息。	atTextGet	TInMessageClient.Get
3	OnGetFiles	取离线消息文件列表。	atTextGetFiles	TInMessageClient.GetMsgFiles
4	OnPush	推送（发消息给其他客户端）。	atTextPush	TInMessageClient.SendMsg 带目的客户端名称
5	OnReceiver	服务端收到普通消息。	atTextSend	TInMessageClient.SendMsg

表 9、TInMessageClient 组件的事件

编号	事件名称	事件说明
1	OnListFiles	列出服务端返回的文件名称（作 TMessagePack 的宿主、提交 atTextGetFiles 请求后返回数据时触发）。
2	OnReturnResult	发出各种消息类请求后，服务端返回数据时触发的事件。

## 七、文件服务

InIOCP 已实现部分文件操作的代码，一些操作要由用户结合实际情况自行实现，下面的介绍出现了 2.0 版的新特性 TMessagePack 的用法，以后会单独对其作介绍。

下表是 TInFileManager 组件的事件：

表 10、TInFileManager 组件的事件

编号	事件名称	事件说明	对应客户端操作	对应客户端组件方法
1	AfterDownload	文件下载完毕时触发的事件（支持断点下载）。	atFileDownload atFileDownChunk	TInFileClient.Download TMessagePack.Post(atFileDownChunk)
2	AfterUpload	文件上传完毕时触发的事件（支持断点下载）。	atFileUpload atFileUpChunk	TInFileClient.Upload TMessagePack.Post(atFileUpChunk)
3	BeforeDownload	下载文件前触发的事件（支持断点下载）。	atFileDownload atFileDownChunk	TInFileClient.Download TMessagePack.Post(atFileDownChunk)
4	BeforeUpload	上传文件前触发的事件（支持断点下载）。	atFileUpload atFileUpChunk	TInFileClient.Upload TMessagePack.Post(atFileUpChunk)
5	OnDeleteDir	删除目录时触发的事件。	atFileDeleteDir (下版本提供)	TMessagePack.Post(atFileDeleteDir)
6	OnDeleteFile	删除文件时触发的事件。	atFileDelete	TInFileClient.Delete TMessagePack.Post(atFileDelete)
7	OnMakeDir	新建目录时触发的事件。	atFileMakeDir (下版本提供)	TMessagePack.Post(atFileMakeDir)
8	OnQueryFiles	查询当前工作目录的文件时触发的事件。	atFileList	TInFileClient.ListFiles TMessagePack.Post(atFileList)
9	OnRenameDir	目录改名时触发的事件。	atFileRenameDir (下版本提供)	TMessagePack.Post(atFileRenameDir)
10	OnRenameFile	文件改名时触发的事件。	atFileRename	TInFileClient.Rename TMessagePack.Post(atFileRename)
11	OnSetWorkDir	设置当前工作目录时触发的事件。	atFileSetDir	TInFileClient.SetDir TMessagePack.Post(atFileSetDir)
12	OnShareFile	共享文件时触发的事件。	atFileShare	TInFileClient.Share TMessagePack.Post(atFileShare)
13	OnTransmitRequest	请求对方接收文件时触发的事件。	atFileRequest	暂未实现

表 11、TInFileClient 组件的事件

编号	事件名称	事件说明
1	OnListFiles	列出服务端返回的文件名称时触发的事件。
2	OnReturnResult	发出各种消息类请求后，服务端返回数据时触发的事件。

## 八、自定义服务

以上几大服务的特点就是作用单一，服务端和客户端的操作相对固定，灵活性不足，扩展不方便。

自定义服务正是为了解决扩展性问题而设计的，它其实是自定义消息服务，它只负责传输消息，不对这些消息的用途作任何解释和处理。

为了扩展功能，用户可以为客户端的自定义消息增加特殊含义的字段（变量/参数），服务端根据它的值判断、执行相应的操作，最后返回结果；当然，这种功能扩展完全可以用远程函数的方法解决。

顺便提一下，InIOCP 的消息分主体和附件，有需要的请参考《InIOCP 技术要点》的基础知识部分。

下面先看自定义服务的 TInCustomManager 组件的事件：

表 12、TInCustomManager 组件的事件

编号	事件名称	事件说明	对应客户端操作	对应客户端组件方法
1	OnAttachBegin	客户端传来的消息带附件时触发的事件，在此判断是否要接收。	atCustomAction (自动设置)	TInCustomClient.Post, 带 LoadFromFile、LoadFromStream
2	OnAttachFinish	消息附件接收完时触发的事件。	atCustomAction (自动设置)	TInCustomClient.Post
3	OnReceive	收到客户端的消息主体后触发的事件（先于 OnAttachBegin 执行）	atCustomAction (自动设置)	TInCustomClient.Post

客户端组件 TInCustomClient 的介绍略，远程函数的使用方法也比较简单，这里也不作细述。

## 九、新特性 TMessagePack