



顶点 ABOSS2

接入端开发文档

福建顶点软件股份有限公司

2007 年 5 月

发布日期	版本号	修改记录
2007.05.09	2.0.0.1	创建发布
2007.05.23	2.0.0.2	1、修正文档中 Fix_RegisterSvc 中的回调函数的参数错误, 改函数用于处理银行发起的业务请求。 2、增加异步函数 Fix_AsyncRun、Fix_IsReplied

第三方接口的文件清单：

文件名称	版本号	说明
FixApi.dll	2.0.0.2	第三方接入接口动态库
FixApi.lib	2.0.0.2	第三方接入接口连接库
FixApi.h	2.0.0.1	第三方接入接口头文件
Libeay32.dll	0.9.8.0	OpenSSL Shared Library
Ssleay32.dll	0.9.8.0	OpenSSL Shared Library
MSVCP60.dll	6.2.3104.0	Microsoft (R) C++ Runtime Library
TestApi.rar		开发包 Demo
FixApi.pdf		开发技术说明文件

概要说明：

系统提供基于 TCP 接入的接口方式, 并提供基于 Fix 的 tag 标识符来表示具体的数据信息。(其中具体的 tag 值由顶点其他业务文档实现) 系统内部实现自动重连服务器的功能。针对系统内部需要通过文件交互进行对账的业务, 系统提供文件上传下载的函数来实现。对于日间银行端发起的业务, 系统提供服务注册(注入)的方法, 通过回调函数来实现。

系统实现两个句柄类型, 分别是连接句柄和会话句柄, 用户使用该连接句柄分配出相应的会话句柄用于一次的业务请求交互过程。特别要注意的是由应用层分配的句柄, 必须有应用层主动释放, 以防止内存泄露。

为了方便用户进行的快速开发, 系统提供基于 C++ 的 Demo 代码。

考虑到系统的性能, 建议使用长连接进行数据的交互。

函数说明

1、Fix_Initialize 接口库初始化

函数名称	Fix_Initialize 第三方接口库初始化。只有调用一次就可以，不能多次调用。
返回数据	返回True表示初始化成功；False表示失败。
函数原型	BOOL Fix_Initialize()

2、Fix_Initialize 接口库卸载

函数名称	Fix_Uninitialize 第三方接口库卸载。只能调用一次就可以，不能多次调用。
返回数据	返回True表示初始化成功；False表示失败。
函数原型	BOOL Fix_Uninitialize ()

3、Fix_SetDefaultInfo

函数名称	Fix_SetDefaultInfo 设置每个业务请求包缺省的头信息。
参数说明	pszUser -- [in] 系统要求的柜员代码(八位的字符串)。 pszWTFS -- [in] 委托方式。 pszFBDM -- [in] 发生营业部的代码信息(四位的字符串) pszDestFBDM -- [in] 目标营业部的代码信息(四位的字符串)
返回数据	返回True表示初始化成功；False表示失败。
函数原型	BOOL Fix_SetDefaultInfo (const char *pszUser, const char *pszWTFS, const char *pszFBDM, const char *pszDestFBDM)

4、Fix_Connect

函数名称	Fix_Connect 连接到顶点中间件服务器。
参数说明	pszAddr -- [in]连接的服务器地址；格式为：“ip地址@端口/tcp” pszUser -- [in] 通信用户名称；由客户提供。 pszPwd -- [in] 通信用户的密码；由客户提供。 nTimeOut -- [in] 连接超时的时间。单位为秒。
返回数据	系统返回类型为HANDLE_CONN的连接句柄。如果连接失败则返回0；成功不为0；
函数原型	HANDLE_CONN Fix_Connect (const char *pszAddr, const char *pszUser, const char *pszPwd, long nTimeOut)

5、Fix_Close

函数名称	Fix_Close 与顶点中间件连接关闭。
------	--------------------------

参数说明	conn -- [in] 类型为HANDLE_CONN的连接句柄。该句柄由Fix_Connect生成的。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_Close(HANDLE_CONN conn)

6、Fix_AllocateSession

函数名称	Fix_AllocateSession 根据一个连接句柄分配一个会话句柄，用于数据交互。由用户自己在应用层通过调用Fix_AllocateSession分配出来的对象必须由用户自己调用Fix_ReleaseSession来释放。反之则不必要。原则上遵循“谁分配谁释放”。这样就可以确保内存不泄露。
参数说明	conn -- [in] 类型为HANDLE_CONN的连接句柄。该句柄由Fix_Connect生成的。
返回数据	返回值类型为HANDLE_SESSION的会话对象；如果对象值为空表示对象分配失败。否则表示成功。
函数原型	HANDLE_SESSION Fix_AllocateSession(HANDLE_CONN conn)

7、Fix_ReleaseSession

函数名称	Fix_ReleaseSession 释放会话句柄。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话对象。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_ReleaseSession(HANDLE_SESSION sess)

8、Fix_SetTimeOut

函数名称	Fix_SetTimeOut 设置会话交互的超时时间。
参数说明	conn -- [in] 类型为HANDLE_CONN的连接句柄。该句柄由Fix_Connect生成的。 timeout -- [in] 业务应答超时时间；单位为秒，系统默认为30秒。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_SetTimeOut (HANDLE_SESSION sess, long timeout)

8、Fix_SetWTFS

函数名称	Fix_SetWTFS 设置会话的委托方式。(必须在Fix_CreateHead函数之前调用)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 Val -- [in] 字符串类型。用于表示客户的接入方式；比如

	电话委托，磁卡委托，互联网委托等。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_SetWTFS (HANDLE_SESSION sess, const char *val)

9、Fix_SetFBDM

函数名称	Fix_SetFBDM 设置会话的发生营业部代码。(必须在Fix_CreateHead函数之前调用)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 Val -- [in] 字符串类型。用于表示客户业务发生的营业部代码, 必须是四位的营业部代码;
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_SetFBDM (HANDLE_SESSION sess, const char *val)

10、Fix_SetDestFBDM

函数名称	Fix_SetDestFBDM 设置会话的目标营业部代码。(必须在Fix_CreateHead函数之前调用)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 Val -- [in] 字符串类型。用于表示客户业务发生的营业部代码, 必须是四位的营业部代码;
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_SetDestFBDM (HANDLE_SESSION sess, const char *val)

11、Fix_SetNode

函数名称	Fix_SetNode 设置会话的业务发生的节点信息。(必须在Fix_CreateHead函数之前调用)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 Val -- [in] 字符串类型。一般是客户网卡信息或IP地址。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_SetNode (HANDLE_SESSION sess, const char *val)

12、Fix_SetGYDM

函数名称	Fix_SetGYDM 设置会话的业务发生的柜员代码信息。(必须在
------	--------------------------------------

	Fix_CreateHead函数之前调用)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 Val -- [in] 字符串类型。一般是客户网卡信息或IP地址。
返回数据	返回值为True表示成功; FALSE表示失败。
函数原型	BOOL Fix_SetGYDM (HANDLE_SESSION sess, const char *val)

13、Fix_CreateHead

函数名称	Fix_CreateHead 设置会话的业务功能号. 具体的功能号的定义请参照【第三方接入业务接口文档】
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 Val -- [in] 整型。即为系统提供的业务功能号。
返回数据	返回值为True表示成功; FALSE表示失败。
函数原型	BOOL Fix_CreateHead (HANDLE_SESSION sess, long nFunc)

14、Fix_SetItem(字符串)

函数名称	Fix_SetItem 设置该会话要发送给中间件的业务的请求域数据 或服务应答域数据。注:(服务应答域数据只是针对一些中间件业务需要第三处理机处理的情况下使用。而且是要求通过注册服务实现; 一般的情况下不使用该方式接入。)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 请求域的tag值; 具体的定义值请参考【第三方接入业务接口文档】。 val -- [in] 字符串类型; 对应于id的业务数据。
返回数据	返回值为True表示成功; FALSE表示失败。
函数原型	BOOL Fix_SetItem (HANDLE_SESSION sess, long id, const char *val)

15、Fix_SetItem(整型)

函数名称	Fix_SetItem 设置该会话要发送给中间件的业务的请求域数据 或服务应答域数据。注:(服务应答域数据只是针对一些中间件业务需要第三处理机处理的情况下使用。而且是要求通过注册服务实现; 一般的情况下不使用该方式接入。)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 请求域的tag值; 具体的定义值请参考【第三方接入业务接口文档】。 val -- [in] 整型; 对应于id的业务数据。

返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	<pre> BOOL Fix_SetItem (HANDLE_SESSION sess, long id, long val) </pre>

16、Fix_SetItem(浮点)

函数名称	Fix_SetItem 设置该会话要发送给中间件的业务的请求域数据 或服务应答域数据。注:(服务应答域数据只是针对一些中间件业务需要第三处理机处理的情况下使用。而且是要求通过注册服务实现; 一般的情况下不使用该方式接入。)
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 请求域的tag值; 具体的定义值请参考【第三方接入业务接口文档】。 val -- [in] 浮点类型; 对应于id的业务数据。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	<pre> BOOL Fix_SetItem (HANDLE_SESSION sess, long id, double val) </pre>

17、Fix_Run

函数名称	Fix_Run 提交业务数据请求, 并等待全部的业务结果返回。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。
返回数据	返回值为True表示服务业务处理成功; FALSE表示失败, 这个失败是表示业务通信上的失败; 失败后, 可以通过Fix_GetCode取出错误码(必定是一个负数)。通过Fix_GetErrMsg取出错误的信息
函数原型	<pre> BOOL Fix_Run(HANDLE_SESSION sess) </pre>

Fix_AsyncRun

函数名称	Fix_AsyncRun 异步提交业务数据, 不等待全部的业务结果返回
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。
返回数据	返回值为True表示服务业务处理成功; FALSE表示失败, 这个失败是表示业务通信上的失败; 失败后, 可以通过Fix_GetCode取出错误码(必定是一个负数)。通过Fix_GetErrMsg取出错误的信息
函数原型	<pre> BOOL Fix_AsyncRunHANDLE_SESSION sess) </pre>

Fix_IsReplied

函数名称	Fix_IsReplied
------	----------------------

	异步提交业务数据后，检查数据是否全部返回。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 nMSec -- [in] 单位秒，等待时间，如果等于零，立即返回。
返回数据	返回值为True表示应答数据全部返回；FALSE表示需要继续等待。
函数原型	BOOL Fix_IsReplied(HANDLE_SESSION sess, long nMSec)

18、Fix_Cancel

函数名称	Fix_Cancel 直接放弃业务应答数据，结束在Fix_Run中的等待。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。
返回数据	返回值为True表示成功；FALSE表示失败。
函数原型	BOOL Fix_Cancel(HANDLE_SESSION sess)

18、Fix_GetCode

函数名称	Fix_GetCode 读取业务提交失败的错误代码。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。
返回数据	返回整数。
函数原型	BOOL Fix_GetCode(HANDLE_SESSION sess)

19、Fix_GetErrMsg

函数名称	Fix_GetErrMsg 读取业务提交失败的错误信息。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 out -- [in/out] 用于输出错误信息的字符串。 outlen -- [in] 参数out缓冲区的大小
返回数据	返回错误信息字符串。
函数原型	char *Fix_GetErrMsg (HANDLE_SESSION sess, char *out, int outlen)

20、Fix_GetCount

函数名称	Fix_GetCount 读取中间件返回的业务应答数据的行数。
参数说明	Sess -- [in] 类型为HANDLE_SESSION的会话句柄。
返回数据	返回行数量。
函数原型	long Fix_GetCount(HANDLE_SESSION sess)

21、Fix_GetItem

函数名称	Fix_GetItem
------	-------------

	1、从中间件的返回结果集的指定行中读取业务应答数据。 2、或者是读取中间件请求的业务请求。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 指定域的tag值；具体的定义值请参考【第三方接入业务接口文档】。 out -- [in/out] 用于输出业务信息的字符串。 outlen -- [in] 指定out缓冲区的大小。 row -- [in][option] 指定返回数据结果集中的行数(行数索引值以0开始). 如果不设置这个值，表示读取第一行的数据。
返回数据	返回指定行的的字符串数据。
函数原型	char *Fix_GetItem (HANDLE_SESSION sess, long id, char *out, int outlen, long row = -1)

22、Fix_GetLong

函数名称	Fix_GetLong 1、从中间件的返回结果集的指定行中读取业务应答数据。 2、或者是读取中间件请求的业务请求。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 指定域的tag值；具体的定义值请参考【第三方接入业务接口文档】。 row -- [in][option] 指定返回数据结果集中的行数(行数索引值以0开始). 如果不设置这个值，表示读取第一行的数据。
返回数据	返回指定行的整型数据。
函数原型	long Fix_GetLong (HANDLE_SESSION sess, long id, long row = -1)

23、Fix_GetDouble

函数名称	Fix_GetDouble 1、从中间件的返回结果集的指定行中读取业务应答数据。 2、或者是读取中间件请求的业务请求。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 指定域的tag值；具体的定义值请参考【第三方接入业务接口文档】。 row -- [in][option] 指定返回数据结果集中的行数(行数索引值以0开始). 如果不设置这个值，表示读取第一行的数据。
返回数据	返回指定行的整型数据。
函数原型	long Fix_GetDouble

	(HANDLE_SESSION sess, long id, long row = -1)
--	---

24、Fix_HaveItem

函数名称	Fix_HaveItem 1、检查中间件的返回结果集的指定行中是否含有指定域。 2、或者检查中间件请求的业务数据是否含有指定域。
参数说明	sess -- [in] 类型为HANDLE_SESSION的会话句柄。 id -- [in] 指定域的tag值；具体的定义值请参考【第三方接入业务接口文档】。 row -- [in][option] 指定返回数据结果集中的行数(行数索引值以0开始). 如果不设置这个值，表示读取第一行的数据。
返回数据	返回True表示, 含有指定的域数据；否则没有。
函数原型	long Fix_HaveItem (HANDLE_SESSION sess, long id, long row = -1)