IPC配置手册

1. **IPC介绍**

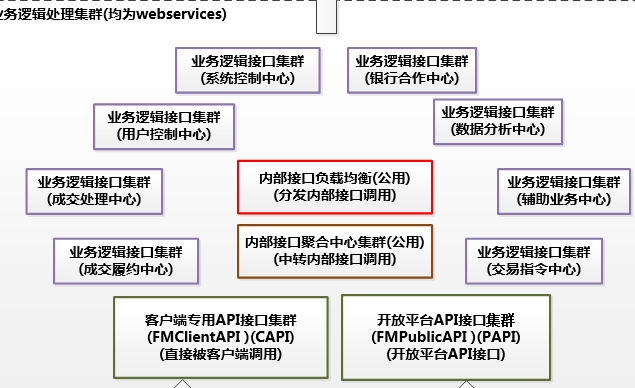
基于webservices建立多组业务服务，每组业务独立成为“xxxx业务处理中心”，同时，每组webervices服务被配置到由多个主机组成的负载均衡环境中。以此完成细分业务级别的负载。

当客户端、web展示层进行业务处理时，通过调用对应接口，实现相关功能。由于业务拆分精细程度可能较高，每个具体功能（远程方法）之间会存在复杂的相互调用关系。

因此，需要一个独立存在的接口管理中心，这就是IPC。

IPC中包含接口具体执行调用的类库。包括接口与接口内方法的维护功能。包含接口关系的管理维护功能。包含对各业务处理中心所在集群中的单一服务器上所部属接口的关系订阅更新的控制功能。

架构图如下：



1. **IPC可解决的问题**

1.所有接口的历次被调用，均可开启详细日志，便于进行精细跟踪。

2.允许异步、同步接口的各种组合调用。某些场景下对相应时间提升明显。

3.在接口调用意外失败后，可以附加自定义的回滚机制。

4.及时发现不通畅的接口。

5.快速查阅每个接口的详细作用和方法。

6.对所有各系统的接口调用语法，变得完全统一。

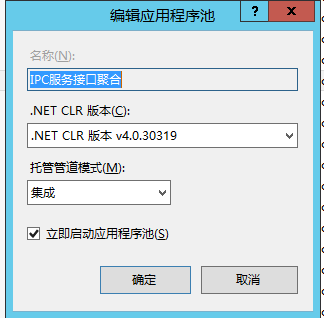
7.各程序开发时，无需再进行webservice的引用。甚至可以忽略接口的存在。

8.所有接口之间，用于业务的互相调用必须通过IPC进行订阅。

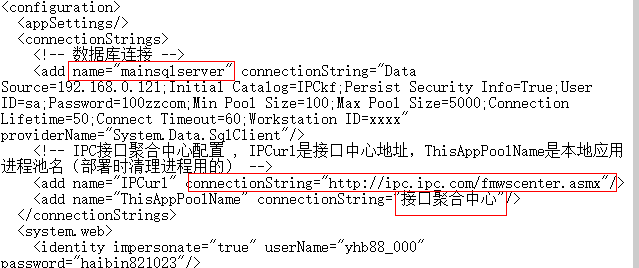
9. 有利于后期架构调整和拓展。

10. 方便分析各业务集群的负载容量，以便于寻找瓶颈合理调整架构。

1. **系统程序部署与配置调整**
2. 将“IPC服务接口聚合”项目，部属至IIS服务器(需管理员权限)，绑定域名ipc.ipc.com，端口80。该域名只在服务器集群内部解析。 应用程序进程池设置如下：



1. 将“Web(样例).config”复制一份拷贝，并将拷贝改名为“Web.config”。关键配置如下：



数据库连接的name，不允许更改。 若需连接其他数据库，需要新增配置。

IPCurl，指IPC的配置中心服务。通常情况不需要修改。

1. **接口关系管理**
2. 在浏览器输入： <http://ipc.ipc.com/IPCadmin>
3. 首先添加新接口和方法：



其中：

接口域名，可以带端口。不能带最后的斜线。没有http://。

接口地址，不带域名信息。不能带最前面的斜线。

录入以上两项后，点击检查，会自动检测接口的其他信息。

选择“有插入或更新操作”，将忽略关系中的日志是否开启，会强制开启日志，但同步异步选项仍然有效。

选择“仅查询操作”，会忽略关系中的调用方式设置，会强制使用同步调用，但日志是否开启选项仍然有效。(因为这种通常是为了查询一些东西，所以没有必要异步)。

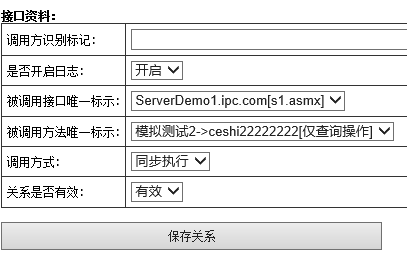
业务名称，尽可能简短，可以使用中文，且与系统其他所有业务不允许重复。将用于代码调用。

方法名称，不能带最后的括号。

返回值类型，不能没有返回值。

1. **添加接口调用关系**

用于建立某个业务服务中心与其他接口内业务方法的对应可调用关系。如下图：

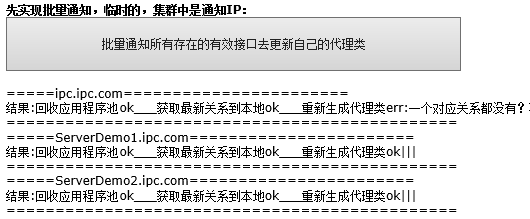


“调用方识别标记”，web.config中配置的名称，需要与应用进程池名称相同。用于记录这条关系是哪个服务在用。

调用方式中，“异步”与“同步”的区别如下：同步调用，系统将等待业务接口内业务完全处理成功，才继续执行后续代码；异步调用，系统将只保存执行指令，不执行而立刻返回，由异步调用处理服务不间断的处理被保存的指令,可以理解为延后执行。

1. **关系订阅**

当各接口关系维护完成后，IPC仅在数据库中进行了记录，并未生效。需要向各业务中心接口发送指令，要求其更新订阅关系，订阅成功后才生效。



订阅成功后，业务处理中心才可以真正使用新的接口关系。接口互相调用时，不再连接IPC，而是通过本地关系和代理类直接互相访问。 但是，开启了日志的接口，每次调用均连接IPC，用于记录日志。

后续完善后，发送更新通知，应指定IP地址，而不是域名。以便于分别按需要进行更新。

1. **业务中心接口编写与调用注意事项**
2. 要求所有接口都引用聚合中心配套的动态链接库。(FMipcClass.dll和FMDBHelperClass.dll)。直接放在Bin目录下即可。不需要添加引用。
3. 业务接口的站点代码，需要使用“新建站点”类型。也就是不需要编译直接运行那种。
4. 业务接口必须存在ForIPC目录。必须放入一个公共webservice(OrderFormIPC.asmx)。这个是特殊接口，用于接收来自中心的指令。不需要配置到聚合中心关系库中。
5. 业务处理中心所在站点的运行帐户，必须具备管理员权限。在webconfig中配置。
6. 每个业务接口内的每个独立接口，都只能有一个类，不允许在接口发布类中编写业务逻辑。必须另外建立类库，接口发布类只是引用。
7. 接口的文件名，必须跟类名完全一样，包括大小写。 因此，尽量都是用小写。
8. 接口内只需要存在方法即可。不需要定义属性。 但方法必须同时具备传入参数和返回值，不允许无参数方法或无返回值的方法。
9. 注意，若方法执行后，返回了null，将被判定为接口存在问题或者程序出现错误 。
10. 方法的参数和返回值，至少应有一个参数。
11. 关于接口返回值的强制约定：

若发生程序错误，并被捕捉到，直接返回null。 具体错误记录日志。

未发生错误，具体返回值以开发人员规定的格式为准。

1. 接口必须编写注释。放到名空间和方法的Description上。例如：



其中，整个接口的说明中，必须带有版本号。用->隔开。

1. 调用例子：

