# U8 EAI接口

## 摘要

U8企业应用集成（EAI）产品是U8ERP与第三方系统进行业务数据交换的平台级工具产品，是U8ERP统一的对外开放接口。

EAI产品包括一整套U8ERP的标准业务数据交换接口，这些接口涵盖基础档案、财务、供应链、生产制造、客户关系管理、HR等ERP业务模块，具备统一的入口、一致的业务外观和相对完整的业务功能，能完成U8基础数据、凭证、业务单据的导入导出及增、删、改查等操作，是U8与外部系统进行应用集成和业务整合的基本资源。

在技术上，EAI数据交换接口采用符合业界标准的XML作为数据交换接口规范（协议）描述，充分利用了XML跨平台和低耦合的特性。基于XML的交换协议描述了数据交换过程中Request/Response消息数据格式及元数据规范（Schema），外部系统只要遵循此标准和规范即可以方便的与U8ERP进行Talking。

为解决数据交换中系统之间数据格式和数据语义差异问题，EAI产品提供了可扩展的数据适配器和数据映射机制用来进行内外数据的翻译和转换；为方便用户的应用和体验，EAI产品还提供了能直接进行U8数据导入导出的基本用户界面工具，此工具支持手工和自动导入导出。

## 目标

本文主要介绍如何通过EAI接口实现U8与其他业务系统之间的数据交互。

## EAI接口规范

### EAI接口

#### 本地COM接口

类库：U8Distribute.DLL

ProgID：U8Distribute.iDistribute

方法：String Process(String RequestXml)

参数：数据交换请求消息，详见下文请求消息格式。

返回：数据交换响应消息，详见下文回执消息格式。

这种方式可根据开发者的需要批量处理或者实时处理。接口调用非常简单，简化了编程。开发者只需将标准数据传入接口，如下：

1. VB调用代码
2. .NET调用代码

//引用U8SOFT\EAI\U8Distribute.dll

U8Distribute.iDistributeClass eaiBroker =

new U8Distribute.iDistributeClass(); //创建EAI服务代理接口对象

String responseXml =

eaiBroker.Process(requestXml); //调用EAI服务代理的数据交换方法Process，传入 // Request交换消息， 并获取EAI返回的Response消息。

...... //处理返回结果

System.Runtime.InteropServices.Marshal.FinalReleaseComObject(eaiBroker); //释放EAI服务代理接口对象

Dim eaiBroker As Object

Set eaiBroker = CreateObject(“U8Distribute.iDistribute”) ’创建EAI服务代理接口对象

Dim responseXml As String

responseXml = eaiBroker.Process(requestXml) ’调用EAI服务代理的数据交换方法Process，传入 ‘Request交换消息， 并获取EAI返回的Response消息。

**….. ’处理返回结果**

Set eaiBroker = Nothing ’释放EAI服务代理接口对象

1. JAVA代码

采用JNI、JCOM、JACOB技术调用EAI服务代理COM组件，下面以JACOB为例：

import com.jacob.com.Dispatch;

import com.jacob.com.Variant;

class Test

{

　 public static void main(String[] args)

{

//创建EAI服务代理接口对象

　　 Dispatch eaiBroker = new Dispatch("U8Distribute.iDistribute");

//调用EAI服务代理的数据交换方法Process，传入Request交换消息， 并获取EAI

//返回的Response消息。

String responseXml =

Dispatch.call(eaiBroker, "Process", new Variant(requestXml));

...... //处理返回结果

}

}

#### 远程HTTP接口

EAI远程数据交换采用Http POST的方式。

访问网址： [**http：//U8应用服务器IP或机器名/U8EAI/import.asp**](http://u8应用服务器ip或机器名/U8EAI/import.asp)。

POST上传的数据交换请求消息与本地调用一样。

开发者可以用SOAP协议向装有U8-EAI的IIS服务器发送XML数据，类似本地调用分析返回的XML回执，如下：

1. VB调用代码

注意： URL： http：//u8appsrv/u8eai/import.asp 中的u8appsrv应填U8应用服务器IP或机器名。

Dim xmlhttp As Object

Set xmlhttp = CreateObject(“MSXML2.XMLHTTP”) ‘创建XMLHTTP对象

Call xmlhttp.open(“POST”, “http://u8appsrv/u8eai/import.asp”, False) ‘使用XMLHTTP对象打开网络连接

xmlhttp.send requestXml ‘发送Request消息

Dim responseXml As String

responseXml = xmlhttp.responseText ‘获取返回Response消息

….. ’处理返回结果

1. .NET调用代码
2. JAVA调用代码

//引用U8SOFT\Interop\Interop.MSXML2.dll

MSXML2.XMLHTTPClass xmlHttp = new MSXML2.XMLHTTPClass(); //创建XMLHTTP对象

xmlHttp.open(“POST”, “<http://u8appsrv/u8eai/import.asp>”, false, null, null);//打开网络连接

xmlHttp.send(dom.OuterXml);//发送Request消息

String responseXml = xmlHttp.responseText;//获取返回Response消息

...... //处理返回结果

System.Runtime.InteropServices.Marshal.FinalReleaseComObject(xmlHttp); //COM释放

#### 调用U8API接口

class Test

{

　 public static void main(String[] args)

{

URL url = new URL(“<http://u8appsrv/u8eai/import.asp>”);

HttpURLConnection con = (HttpURLConnection)url.openConnection();

con.setConnectTimeout(3000000);

con.setReadTimeout(3000000);

con.setDoInput(true);

con.setDoOutput(true);

con.setAllowUserInteraction(false);

con.setUseCaches(false);

con.setRequestMethod("POST");

con.setRequestProperty("Content-type","application/x-www-form-urlencoded");

//发送Request消息

OutputStream out = con.getOutputStream();

DataOutputStream dos = new DataOutputStream(this.out);

dos.write(requestXml.getBytes("UTF-8"));

//获取Response消息

InputStream in = con.getInputStream();

BufferedReader br = new BufferedReader(new InputStreamReader(in));

StringBuilder sb = new StringBuilder();

String s = null;

while ((s = br.readLine()) != null) {

sb.append(s);

}

String responseXml = sb.toString();

}

}

满足现有U8EAI做不到的业务单据操作，如销售报价单增删改，以及业务操作，如审核、弃审，将现有U8API的功能整合进U8EAI，将两者糅合在一起，形成功能互补，为企业应用集成和二次开发提供更强大的支持。

通过参照U8API请求XML模板，填入相应的API参数值，即可实现通过U8EAI统一接口调用U8API服务。详细内容请参见文档**[《 U8API](《%20U8API和U8EAI整合》)**[**和U8EAI整合》**](《%20U8API和U8EAI整合》)。

1. **请求XML格式**

<?xml version="1.0" encoding="utf-8" ?>

<ufinterface roottag="**u8api**" receiver="**u8**" sender="**001**" proc="**add**">

<u8apiservice>

<apiurl value="**U8API/PurchaseOrder/VoucherSave**" />

<apiParams>

<param name="**VoucherState**" value="**2**" datatype="**int**" />

<param name="**CurDom**" datatype="**IXMLDOMDocument2**" />

<param name="**UserMode**" value="**0**" datatype="**int**" />

</apiParams>

<bo>

<head type="**bo**"/><body type="**bo**“/>

</bo>

<returnParams>

<param name="**curID**" />

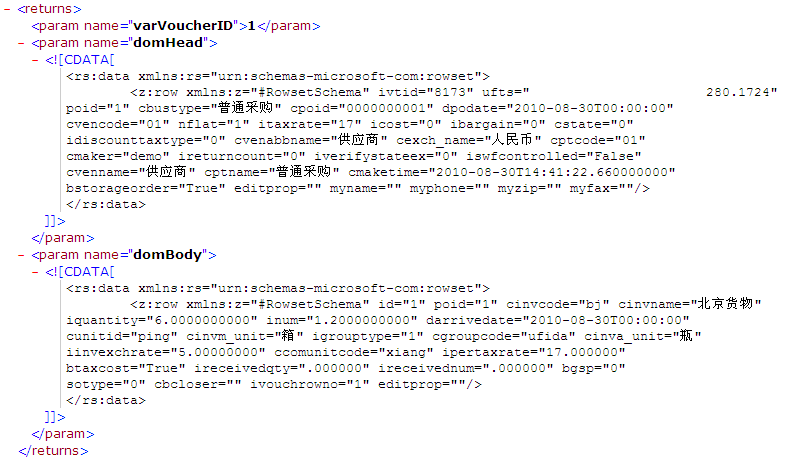
</returnParams>

</u8apiservice>

</ufinterface>

其中，proc填入操作类型，添加：add；编辑：edit；删除：delete；查询：query；审核：verify；弃审：unverify。

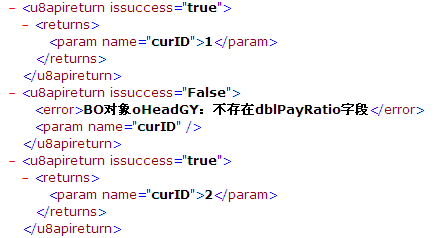
1. **返回XML格式**



以上为查询返回XML格式，新增成功后，还会返回新增单据或档案的ID。



1. **返回错误信息**



按传入顺序返回批量操作的执行结果，调用一次API的过程中若出现错误，不影响其他调用的执行。

#### 通用SQL查询接口

由于EAI接口返回的数据格式是固定的标准格式，很有可能返回的数据中没有用户所需要的字段；另外，对于U8中的业务数据，EAI接口也不是100%覆盖的。

针对以上情况，EAI提供了一个灵活的数据查询通用接口--EAI通用SQL查询接口，该接口支持向EAI数据交换代理传递SQL语句进行查询操作，可以查询当前U8帐套的所有数据，用户不用直接访问U8帐套库。查询返回的结果为.NET中的Dataset的XML格式的数据。调用该接口只需给EAI数据交换服务代理传递特殊的Request消息即可。

1. 请求Request消息格式

<?xml version="1.0" encoding="utf-8" ?>

<ufinterface sender="**001**" receiver="**u8**“

roottag="SQLEXE” proc=“department“

codeexchanged=“n” >

<sql value=“select \* from inventory”>

</ufinterface>

其中，框架部分的roottag值必须固定为“SQLEXE”，proc的值用来标识Dataset的行节点，可以是任意字符串；内容部分<sql value=“sql语句”/> 节点为SQL查询语句。

1. 应答Response消息格式

<?xml version="1.0" encoding="utf-8" ?>

<ufinterface sender="**u8**" receiver="**001**"

roottag="SQLEXE" proc="department"

request-roottag="SQLEXE">

<NewDataSet>

<department>

<cDepCode>1</cDepCode>

<bDepEnd>true</bDepEnd>

<cDepName>采购1部</cDepName>

</department>

<department>

<cDepCode>2</cDepCode>

<bDepEnd>true</bDepEnd>

<cDepName>销售1部</cDepName>

</department>

</NewDataSet>

</ufinterface>

其中，根节点的proc的值用来标识Dataset的行节点，一般为表名。

### 数据交换格式

U8EAI提供了一整套完整的基本档案和单据的数据交换模版（在EAI\XML\Template目录下），模板中定义了消息内容部分的格式和字段。

#### 请求消息格式

EAI数据交换请求消息格式分为框架和内容两部分：

框架部分包含数据交换的控制信息，其中主要包括：

<ufinterface sender=“001” receiver=“u8” roottag=“department” proc=“Add” dynamicdate=“12/25/2010”> —**框架部分**

< department>…</department > —**内容部分**

</ufinterface>

1. billtype系统用，可填空；
2. docid：唯一编号，可空；
3. sender: 发送方，填外部系统注册码（必填）；
4. receiver: 接收方，可填U8；
5. roottag: 档案或单据模版名，填档案或单据的唯一标识，如:客商档案：customer，客商分类：customerclass ，具体名称由总体确定，在数据交换中该名称要经常使用；
6. proc：操作类型，分为“增删改查”，对应填Add / Delete /Edit /Query（必填），该字段导入操作，请填写Add / Delete /Edit，导出操作，请填写Query；
7. dynamicdate： 业务账套登录日期，如果不填，则取U8应用服务器所在操作系统的当前日期，否则取指定的登录日期进行业务账套登录；
8. codeexchanged：编码是否已转换，该字段在导入的时候使用，如果已转换即已和U8基础数据编码一致填Y，将不会通过对照表的转换，如果没有转换即和U8基础数据编码不一致填N，将会自动通过对照表转换之后，进行相应的操作；
9. exportneedexch：导出是否需要根据对照表进行转换，需要填“Y”，不需要填“N”，导出的时候使用；
10. timestamp：时间戳标志，在导出数据时，导出此时间戳的所有以后的数据，若为空，则导出只有过滤条件的数据。

内容部分：指数据请求或响应消息中携带的档案或单据等业务数据，如：

<department>  
  <code>1</code>  
  <endflag>0</endflag>  
  <name>综合处</name>  
  <rank>1</rank>  
  <manager />  
  <prop>管理兼技术</prop>  
</department>

单据模板指的U8中各个基础档案和业务档案用XML存储时的格式。目前放在U8Soft\EAI\XML\Template中。模板的下部注释中有模板中各个节点的相应说明，主要是供二次开发人员和客户了解U8单据格式使用。

在U8SOFT\EAI\XML\Samples有各个模板的详细例子供参考。

#### 回执消息格式

从U8导出数据成功的回执就是业务数据。导入情况的回执如下所示：

<ufinterface roottag=“return” docid=“823482692” proc=“add”>  
 <item key="001" succeed="0" dsc="" u8key='001'/>  
 <item key="002" succeed="0" dsc="" u8key='002'/>  
</ufinterface>

返回字段描述：

1. key：单据表体行数据的编号，比如：销售订单表体的存货编码；
2. succeed：成功标识：0：成功；非0：失败；
3. dsc：失败的描述信息；
4. docid：消息号；
5. proc：操作码；
6. u8key：成功后U8系统对应的关键字。

凭证的add操作的Response格式比较特殊，返回格式如下：

<ufinterface roottag="voucheraddreturn" docid="" proc="add">  
  <item accounting\_period="10" voucher\_type="记" voucher\_id="888" u8voucher\_id="1138" entry\_id="" succeed="0" dsc="凭证已成功保存" u8accounting\_period="10" ></item>  
</ufinterface>

返回字段描述：

1. acc\_period：会计期间，0为期初往来明细帐，21为期初待核银行帐，20为银行帐科目调整前余额，1-12为凭证及明细帐
2. vouch\_type：凭证类别字
3. vouch\_id：凭证号(由系统分配凭证号，期初时可为空)
4. row\_id：行号(由系统赋值，期初时为1)
5. u8vouch\_id：U8凭证号
6. u8acc\_period：U8凭证会计期间
7. succeed：成功标识：0：成功；非0：失败；

## EAI接口设置

在使用EAI之前的第一步必须进行一个接口配置。检查有没有进行EAI接口配置（“开始”->“程序”->“用友U8V\*”->“企业应用集成(EAI)”->“EAI接口设置”），如果没有则需要进行配置。一般配置好后就不要动了，除非出现服务器迁移或重新安装的情况。

### 基本设置

需要指定U8服务器、EAI登录身份验证、EAI数据源。如果进行数据交换的数据包比较大，则需要点“自动设置IIS”更改IIS的大数据限制。



建议： EAI的登录身份验证和U8业务帐套的身份验证分开使用。在此将EAI登录用户配置成一个独立用户，不要与其他人使用的账号相同，如demo，并赋予该用户适当的权限以进行EAI数据交换，如帐套主管，配置完后重启IIS。

### 外部系统注册

在“EAI接口设置”中切换到外部系统注册页签进行外部系统注册。可以注册多个外部系统注册码。



在数据交换中，需要明确指定外部系统与U8的某个业务账套的对应关系，该对应关系由一个唯一的注册码标识。可以注册多个外部系统注册码。数据交换时需要在请求消息的头指定一个注册码：<ufinterface sender="注册码" …>。

## 数据导入

### 基础导入

EAI的数据导入，传入的参数为根据XML模板制作的XML数据。比如，要导入部门档案，传入的sXml如下：

|  |
| --- |
| <?xml version="1.0"?>  <ufinterface sender="011" receiver="u8" roottag="department" docid="54811775" proc="Add" codeexchanged="n" exportneedexch="N" paginate="0" display="部门档案" family="" timestamp="0x000000000027B59C">  <department>  <code>0</code>  <endflag>0</endflag>  <name>基建中心</name>  <rank>1</rank>  <manager/>  …  </department>  <department>  <code>001</code>  <endflag>1</endflag>  <name>基建投资部</name>  <rank>2</rank>  <manager/>  <prop/>  ...  </department>  </ufinterface> |

导入时，单据头中的proc必须为Add/Edit/Delete之一，并不是所有单据都支持这三种操作。

### 凭证导入

由于凭证导入的特殊性，凭证导入时单据头中增加了renewproofno、import、newadd三个属性，含义如下：

proc=”add” //追加

proc=”edit” //覆盖

renewproofno=”y” //重新编号

renewproofno=”n” //不重新编号

import=”y” //导入式

newadd=”y” //新增式

并且，导入时只有下面六种组合

|  |  |  |  |
| --- | --- | --- | --- |
| 1. **序号** | * 1. **操作** | * 1. **单据头** | * 1. **说 明** |
| 1. **1** | 1. **重新编号+追加+新增式** | 1. Proc=”add” renewproofno=”y” 2. newadd=”y” 3. import=”n” | 1. 追加生成的凭证记录为生成态。即刚编写完的凭证状态。 2. 凭证编号重新排序 |
| 1. **2** | 1. **不重新编号+追加+新增式** | 1. Proc=”add” renewproofno=”n” 2. newadd=”y” 3. import=”n” | 1. 追加生成的凭证记录为生成态。即刚编写完的凭证状态。 2. 凭证编号不重新排序 |
| 1. **3** | 1. **重新编号+追加+导入式** | 1. Proc=”add” renewproofno=”y” 2. newadd=”n” 3. import=”y” | 1. 保持除记账、预算审批标志外的所有导入前输出状态。 2. 凭证号重新编号 |
| 1. **4** | 1. **不重新编号+追加+导入式** | 1. Proc=”add” renewproofno=”n” 2. newadd=”n” 3. import=”y” | 1. 保持除记账、预算审批标志外的所有导入前输出状态。 2. 凭证号不重新编号 |
| 1. **5** | 1. **不重新编号+覆盖+新增式** | 1. Proc=”edit” renewproofno=”n” 2. newadd=”y” 3. import=”n” | 1. 覆盖生成的凭证记录为生成态。即刚编写完的凭证状态。 2. 凭证号不重新编号 |
| 1. **6** | 1. **不重新编号+覆盖+导入式** | 1. Proc=”edit” renewproofno=”n” 2. newadd=”n” 3. import=”y” | 1. 保持除记账、预算审批标志外的，所有导入前输出状态。 2. 凭证号不重新编号 |

XML是对大小写敏感的，在EAI标准中所有的元素名都为小写字母

注意：EAI标准数据中的有些元素可以为空元素，有些则不得为空，根据具体业务不同而不同。但所规定的元素一个也不能多，一个也不能少

## 数据导出

操作码为Query，可加入导出条件。格式例子（以导出部门档案为例）：

|  |
| --- |
| <?xml version="1.0" ?>  <ufinterface sender="011" receiver="u8" roottag="department" docid="874080836" proc="Query" codeexchanged="n" exportneedexch="n" paginate="0" display="部门档案" family="基础档案">  <department importfile="" exportfile="" code="011" bincrementout="n">  <field display="部门编码" name="cDepCode" operation="=" value="001" logic=" And ( " />  <field display="部门名称" name="cDepName" operation="=" value="001" logic=" ) " />"  </department>  </ufinterface> |

元素department表示为部门档案：

属性importfile：保存的是导入文件路径

属性exportfile：保存的是导出文件路径

属性code：外部系统的注册号

元素Field表示每个条件：

属性display：中文含义

属性name：字段名

属性operation：是比较运算符

属性value：比较值

属性logic：是连接的逻辑运算符。and：并且；or：或者

每次导出一种单据，这里为部门档案，如果需要导出收款单则需要再一次导出。条件Field可以有多条。这个调用返回的XML回执为U8的标准数据。

注意：并不是导出数据会自动保存到Importfile exportfile指定的路径，这两个参数是EAI工具保存数据使用的。

## 应用案例

### 案例背景

A公司是一家专业办公设备经销商。在企业创建之初，由于规模和业务量都不大，在管理上并没有出现多大问题。随着公司的发展，在财务上采用传统手工处理的方式逐渐暴露出越来越多的问题。于是公司引进了一套U8财务系统来有效管理企业的应收账款和出入现金流。

为了更好的拓展和辐射外地市场，提高市场占有率和客户服务水平，公司逐步在全国许多城市建立了自己的经销网点和客户服务中心，为了提高运营效率，降低成本，加强管控，公司决定为整个分销网络部署一套综合业务系统，通过资源共享与实时调度提高整个销售网络的灵活度和有效应对市场变化的能力，并为客户提供一流的服务水平。

但问题随之产生，由于A公司的外地营销网点规模普遍很小，不具备独立核算能力，而且公司即将部署的综合业务系统并不具备专业的财务管理功能，在财务上仍然必须由公司集中管控。而且，多年来，公司在U8财务软件的帮助下已逐步形成了一套严谨高效的财务管理流程和制度，如果摈弃U8，去开发和部署新的能连接业务系统的财务软件，不仅会浪费公司现有的IT资产，增加IT投资成本，更重要的是会浪费公司积累起来的适合自己实情的宝贵管理经验。

A公司老总认为，如果能将综合业务系统与现有的U8财务软件实现整合，必将带来双赢的利益：一方面降低了IT投资成本，节约了时间；另一方面促进了公司财务管理的持续完善和创新。

### 需求分析

A公司的综合业务系统能有效管理经销网点的销售和售后业务帐以及仓库台帐：能开具发票，进行收款结算；能管理发货和收货，进行商品盘点和结存；能统计和分析销售状况；能处理退货、维修等售后业务。

这些业务活动为公司带来的盈利或亏损，必须要通过公司总部的会计系统（U8财务系统）进行记账、核算、以及财务报表分析，才能正确反映公司的经营状况和业绩。

这就是说，综合业务系统和U8财务系统的连接必须要能实现自动记账功能。具体来说就是要将综合业务系统的结算凭证—销售发票转换成U8的记账凭证，并传递给U8系统进行会计记账。

另外，对于客户往来的应收款凭证，一般包含有客户、部门等辅助核算项。而客户、部门、人员等这些基础资料在企业的业务系统和U8财务系统中很有可能不一致，不一致的情况一般有以下几种：

1、A公司的综合业务系统由于直接面向市场客户，故其包含的客户资料比U8财务系统更多；

2、由于两个系统的基础资料编码规则可能不一致，对于同一个客户资料，可能在业务系统中的编码是C-01，在U8系统中可能是001。

以上两种情况的存在，都会导致相关的客户往来凭证无法导入U8记账。

### 解决方案

要实现自动记账功能，首先需要将业务系统的结算凭证—销售发票转换成U8的记账凭证；随后还要能将转换后的记账凭证传递到U8。

解决方案是使用U8EAI数据交换接口。 U8EAI数据交换接口提供了U8与外部系统进行交流的“窗口”，U8EAI定义了一套标准的数据交换格式，外部系统只要遵循此格式，在数据一致的前提下，便能与U8进行交互。

数据一致的意思是要保证各种数据依赖条件的一致，在本项目中就是指凭证中所记录的经济活动核算对象和要素要一致，比如：凭证中所反映的客户、部门等核算对象信息在业务系统和U8中要保持一致，如果不一致，需要进行数据同步或映射转换。

遵循U8的凭证格式是指需要采用某种方式将业务系统的销售发票转换成U8的凭证格式，这一般需要进行二次开发。

在上传凭证之前需要首先保证业务系统和U8的客户档案等基础资料一致。

如果是第一种情况，即客户档案的编码可以相同，但业务系统中的客户多于U8系统中的客户。如果数量不大且不会经常变化，则可以采用手工录入的方式进行补平；如果数量很大且经常变化，则最好采用自动的方式：业务系统中如果增加了客户，则将此客户自动导入到U8系统中，导入的方式是调用U8EAI接口进行客户档案导入，具体实现与凭证自动导入类似，需要遵循U8EAI客户档案数据交换格式。

对于第二种情况，即双方系统的编码规则不一致，如果两边档案完全相同，只是编码不一致，则可以采用编码映射的方式进行对照转换；如果编码不一致的情况下客户档案数量也不相同，业务系统多于U8系统，则需要按第一种情况的做法将客户档案同步到U8，同步的同时还需要进行编码规则转换，这种情况较为复杂，需要找出编码转换规律。

在本案例中只考虑以下两种情况：

1、编码相同情况下，需要将业务系统中的新增客户档案导入U8；

2、客户档案完全一样，只是编码不相同，需要进行编码映射。

### 开发步骤

第一种情况的开发步骤：

1. 将业务系统的客户档案转换成U8客户档案。

U8EAI客户档案的模板文件为：U8SOFT\EAI\XML\Template\Customer.xml，依据此模板文件将外部系统的客户档案格式转换成U8客户格式，可以采用二次开发编程的方式进行，也可以采用比较通用的XSLT进行转换。

在本案例中假定已经做好了客户档案转换，我们只需要事先准备好转换后的客户档案数据即可，准备的方法跟准备凭证相同，可以在U8中建一个客户档案并使用EAI导出。假定文件名为“客户档案.xml”。打开此文件，修改根节点的proc属性值为“add”，即 <ufinterface sender=“001” receiver=“u8” roottag=“customer” proc=“add“ … （示例详见SAMPLES\客户档案.xml ）。

1. 将转换后的客户档案使用U8EAI接口导入U8。

调用代码如下：

Dim oDom As Object

Set oDom = CreateObject("Microsoft.XMLDOM")

Dim xmlContent

xmlContent = "<?xml version='1.0' encoding='utf-8' ?><ufinterface sender='888' receiver='u8' roottag='department' docid='925853908' proc='Query' codeexchanged='N' exportneedexch='N' paginate='0' display='部门档案' family='基础档案' timestamp='0x0000000001077E67'><department><code>0</code><endflag>0</endflag><name>基建中心</name><rank>1</rank><manager /><prop /><phone /><address /><remark /><creditline /><creditgrade /><creditdate /><ddepbegindate>2004-10-28</ddepbegindate><ddependdate /><vauthorizedoc /><vauthorizeunit /><cdepfax /><cdeppostcode /><cdepemail /><cdeptype>中心</cdeptype></department></ufinterface>"

If (oDom.Load(xmlContent)) Then

Dim xmlhttp As Object

Set xmlhttp = CreateObject("MSXML2.XMLHTTP")

Call xmlhttp.open("POST", “http：//u8appsrv/u8eai/import.asp”, False)

xmlhttp.send oDom.documentElement.xml

Dim ret As String

ret = xmlhttp.responseText

'todo: 解析返回结果

Dim retDom As Object

Set retDom = CreateObject("Microsoft.XMLDOM")

If (retDom.loadXML(ret)) Then

retDom.save "c:\EAIReponse.xml"

Else

MsgBox ret

End If

End If

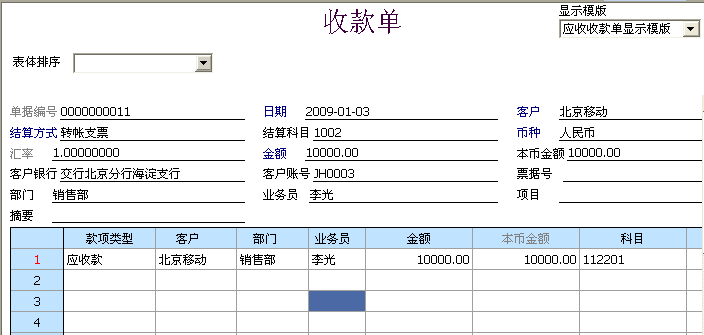
1. 将业务系统结算凭证转换为U8记账凭证。

凭证的模板为Voucher.xml，在此文件中描述了凭证所包含的表头字段以及表体的分录字段、辅助核算字段等，对字段的数据类型、大小、取值规则也做了规定。

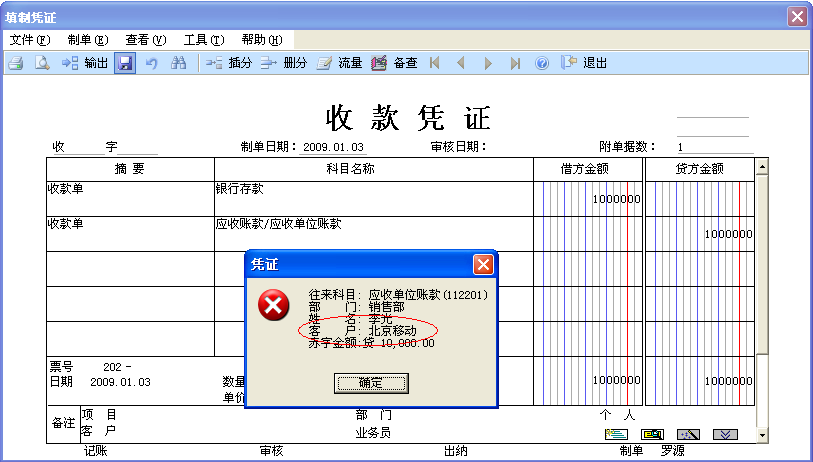
要将外部系统的销售发票格式转换成U8凭证的格式，一般采用二次开发编程的方式进行，也可以采用比较通用的XSLT进行转换。

由于凭证内容应该是带客户辅助核算的应收款凭证，其中客户为前两个步骤导入U8的新客户。

首先，通过应收款系统录入一张收款单：

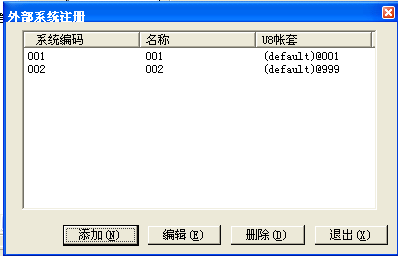


随后，制单生成凭证。

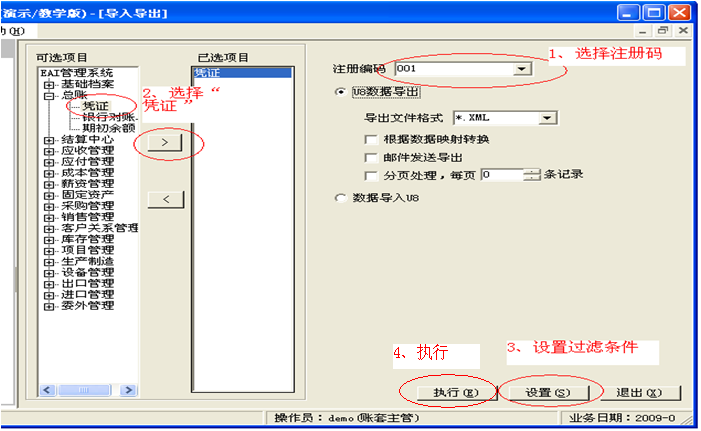


接着，使用EAI将其导出，注意要用刚才建的凭证号设置导出过滤条件。

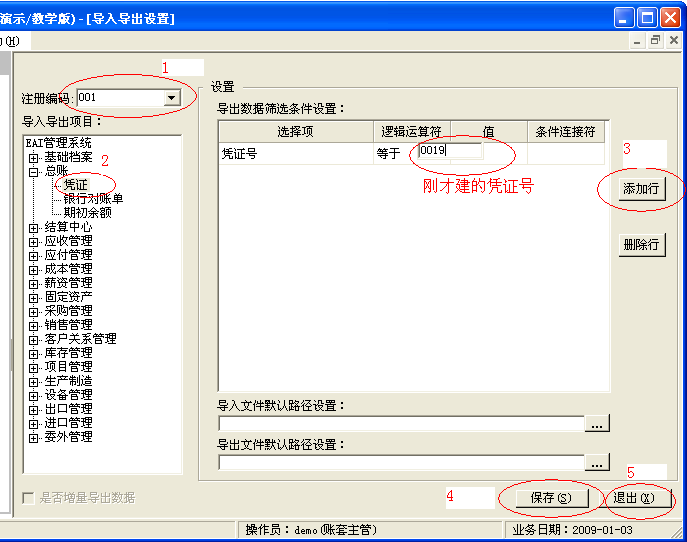
1. 登录EAI：“开始”->“程序”->“用友ERP-U8”->“企业应用集成（EAI）”->“企业应用集成”；
2. 左树“系统菜单”->“设置”-> “系统注册”，为U8的某个帐套注册一个数据交换编码，表明与外部系统的对应关系；



1. 左树“系统菜单”->“数据交换”-> “数据交换”，先设置凭证过滤条件，然后导出为一个XML文件：



1. 设置凭证过滤条件：



点“保存”后“退出”回到上一个界面，点“执行”即可。

假设导出的凭证文件为“简单凭证.xml ”，用编辑工具（比如：记事本）打开此文件，将根节点修改一下：

<ufinterface proc="add" enewproofno="y">

根节点ufinterface的两个属性含义如下：

proc="add" ：表示操作类型，add表示新增。

renewproofno="y"：表示凭证导入U8后是否重新编码，应该填y。

修改完后保存。这样，带辅助核算项的凭证数据准备好后，随后就可以使用EAI数据交换接口将其导入U8并记账。

注：这里导出的凭证分录中就包含客户辅助核算项：

<auxiliary\_accounting>

<item name="dept\_id">1</item>

<item name="cust\_id">C007</item>

…

<auxiliary\_accounting>

4、调用U8EAI接口将凭证传递给U8并记账。

调用EAI接口将凭证导入U8，以VB代码为例：

Dim xmlhttp As Object

Set xmlhttp = CreateObject(“MSXML2.XMLHTTP”) ’创建XMLHTTP对象

Call xmlhttp.open(“POST”， “http：//u8appsrv/u8eai/import.asp”， False) ‘使用XMLHTTP对象打开网络连接，URL中的u8appsrv填U8应用服务器

xmlhttp.send requestXml ’发送Request消息

Dim responseXml As String

responseXml = xmlhttp.responseText ’获取返回Response消息

….. ’处理返回结果

**凭证接口的返回格式如下：**

<ufinterface roottag="voucheraddreturn" docid="" proc="add" sender="u8" receiver="001" request-roottag="voucher">

<item accounting\_period="1" voucher\_type="收" voucher\_id="9" entry\_id="" succeed="0" dsc="凭证已成功保存" u8accounting\_period="1" u8voucher\_id="19"/>

</ufinterface>

其中，succeed=“0”表示成功保存； u8voucher\_id=“19”返回凭证号。

**第二种情况的开发步骤：**

1、在U8EAI中配置客户档案编码映射。

* 登录EAI：“开始”->“程序”->“用友ERP-U8”->“企业应用集成（EAI）”->“企业应用集成”；
* 左树“系统菜单”->“设置”-> “系统注册”，为U8的某个帐套注册一个数据交换编码，建立与外部系统的对应关系；
* 左树“系统菜单”->“数据交换”-> “数据映射”，为U8客户档案与业务系统客户档案设置编码映射关系：



2、将业务系统结算凭证转换为U8记账凭证

同第一种情况的第3步，主要目的是为了制作导入U8的凭证数据。

另外要注意凭证Xml数据的ufinterface节点的控制属性设置：<ufinterface proc="add" enewproofno="y" codeexchanged="y">

根节点ufinterface的三个属性含义如下：

proc="add" ：表示操作类型，add表示新增。

renewproofno="y"：表示凭证导入U8后是否重新编码，应该填y。

codeexchanged="y" ：表示辅助核算项进行编码转换，应该填y。

凭证数据准备好后，就可以调用EAI接口导入了。

3、调用U8EAI接口将凭证传递给U8并记账，在此过程中自动进行客户档案编码转换。调用方法详见第一种情况的步骤4**。**

**以上通过一个完整的案例介绍如何通过EAI接口实现U8与外部业务系统进行数据交换的基本思路和做法。**