**一卡通系统**

**接口API规范**

**bhtm**

|  |
| --- |
| **北航冠新世纪软件有限公司** |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| **版本** | **修订章节** | **修订内容概述** | **修订日期** | **修订人** | **修订前版本号** | **批准人** | **备注** |
| V1.0 |  | 创建 | 2010/10/09 | 袁飞 | 无 |  |  |
|  |  |  |  |  |  |  |  |

目录

[1 引言 4](#_Toc308435827)

[1.1 概述 4](#_Toc308435828)

[1.2 目标 4](#_Toc308435829)

[1.3 接口原则 4](#_Toc308435830)

[1.4 接口调用方式 4](#_Toc308435831)

[1.5 约束与限制 4](#_Toc308435832)

[2 管理规则详细描述 4](#_Toc308435833)

[3 接口详细描述 5](#_Toc308435834)

[3.1 总述 5](#_Toc308435835)

[3.2 接口定义说明 5](#_Toc308435836)

[3.2.1 接口说明 5](#_Toc308435837)

[3.2.2 GetCardVersion 5](#_Toc308435838)

[3.2.3 CardInit 6](#_Toc308435839)

[3.2.4 CardClose 7](#_Toc308435840)

[3.2.5 ReadAnyInfo 7](#_Toc308435841)

[3.2.6 ReadInfo 8](#_Toc308435842)

[3.2.7 QueryInfo 10](#_Toc308435843)

[3.2.8 WriteInfo 11](#_Toc308435844)

[3.2.9 ScanCard 12](#_Toc308435845)

[3.2.10 PatchCard 13](#_Toc308435846)

[3.2.11 PrintCard 14](#_Toc308435847)

[3.2.12 CreateCard 15](#_Toc308435848)

[3.2.13 FormatCard 15](#_Toc308435849)

[3.2.14 CardIsEmpty 16](#_Toc308435850)

[3..15 err 16](#_Toc308435851)

[3.3 附件说明 17](#_Toc308435852)

[3.3.1 卡初始化文件 17](#_Toc308435853)

[3.3.2 卡读写文件 18](#_Toc308435854)

[3.3.3 卡面打印风格文件 19](#_Toc308435855)

[3.3.4 卡面数据文件 19](#_Toc308435856)

1. 引言
   1. 概述

本文当描述了一卡通管理平台在上层需按本API规范接口实现自己的应用程序

* 1. 目标

一卡通管理平台为上层开发提供接口

* 接口简单便于操作,并能够兼容接口以后扩展.
* 提供具有开放的,能够具备准入功能,保证系统的安全性.
  1. 接口原则

上层开发者需严格按照接口规范文档中定义的规则，进行对于卡应用的调用。

* 1. 接口调用方式

提供动态链接库和ATL，方便CS和BS程序开发。

* 1. 约束与限制

本文档适用于一卡通管理系统开发的厂家。

1. 管理规则详细描述
2. 接口详细描述
   1. 总述

实现一套完整的通用的卡应用驱动系统,保证业务系统同卡系统完全解耦,并且能够适应未来卡数据(卡类型)的变更/升级,读卡器的变更/升级.

制定卡应用的标准协议.

* 1. 接口定义说明

### 3.2.1 接口说明

以下是卡应用驱动接口的dll方式描述,卡应用驱动还提供一套activeX方式的接口,接口形式,名字前加入ATL,例如:

动态库方式: iCardInit()

activeX方式: iATLCardInit()

依次类推.

### 3.2.2 GetCardVersion

得到卡版本号，并返回。

int iGetCardVersion(

char \*pszVersion

）;

参数：

pszVersion：用于保存返回的版本号。

返回值：

如果读取卡版本号成功，返回0，否者返回非零错误码。

备注：

pszVersion内存空间在调用程序中分配。

### 3.2.3 CardInit

卡应用驱动初始化，动态绑定制定的卡片描述以及连接硬件的回调函数。

int iCardInit();

参数：

空

返回值：

初始化成功，返回0，否者返回非零错误码。

CardOpen之前必须调用iCardInit,才能开始操作.实现系统初始化

### 3.2.4 CardDeinit

卡应用驱动引擎关闭。

int CardDeinit ( void );

参数：

无；

返回值：

关闭系统成功，返回0，否者返回非零错误码。

备注：

### 3.2.5 CardOpen

打开卡设备。

int iCardOpen( void );

参数：

无；

返回值：

打开成功，返回0，否者返回非零错误码。

备注：

卡设备的各类操作必须首先调用CardOpen,才能开始操作,可重复调用以实现多用户访问卡设备.

### 3.2.6 CardClose

卡设备关闭。

int iCardClose( void );

参数：

无；

返回值：

关闭成功，返回0，否者返回非零错误码。

备注：

无

### 3.2.7 ScanCard

寻卡

int iScanCard (void)；

参数：

返回值：

如果寻卡成功，返回0，否者返回非零错误码。

备注：

无

### 3.2.8 ReadAnyInfo

读取单个或者指定区域的卡数据，并返回XML缓冲区格式的结果。

int iReadAnyInfo(

long iFlag,

char \*pszXml,

char \*pszName

);

参数：

iFlag: 表示卡逻辑的标志：

0x01：制卡信息 ;

0x02：参合基本信息;

0x04: 农合补偿信息;

0x08: 补偿日志;

0x10: 健康档案信息;

其他：待扩展；

pszXml: 用于保存XML格式缓冲区。

pszName:查询单个字段,如果此字段为NULL,则读取指定区域的卡数据,否则按照指定的字段读取.

返回值：

如果调用成功，返回0，否者返回非零错误码。

备注：

XML缓存区至少8k.

本接口返回卡中的一块或者多块或者单个字段数据.

pszXML的空间由调用函数申请

XML文件格式见3.3.2

### 3.2.9 ReadInfo

读取指定区域的卡数据，并返回XML缓冲区格式的结果。

int iReadInfo(

long iFlag,

char \*pszXml

);

参数：

iFlag: 表示卡逻辑的标志：

0x01：制卡信息 ;

0x02：参合基本信息;

0x04: 农合补偿信息;

0x08: 补偿日志;

0x10: 健康档案信息;

其他：待扩展；

pszXml: 用于保存XML格式缓冲区。

返回值：

如果调用成功，返回0，否者返回非零错误码。

备注：

XML缓存区至少8k.

本接口返回卡中的一块或者多块数据.

XML的空间由调用函数申请。

例子:

readBuf = (char \*)malloc(8192);

ret = iReadInfo(0x01|0x02, readBuf);

XML文件格式见3.3.2

### 3.2.10 QueryInfo

读取指定的单个或多个组合，并返回XML缓冲区格式的结果。

int iQueryInfo(

char \*pszName,

char \*pszXml

)；

参数：

pszName: 指定单个或多个字段名称（与卡描述XML中的TARGET保持一致），以|分割多字段。

pszXml: 用于保存XML格式缓冲区。

返回值：

如果调用成功，返回0，否者返回非零错误码。

备注：

XML缓存区至少8k。

XML空间由调用函数申请。

例子：

readBuf = (char \*)malloc(8192);

ret = iQueryInfo(”CARDNO|IDNUMBER“, readBuf);

### 3.2.11 WriteInfo

写入卡片指定XML格式的信息，并返回。

int iWriteInfo(char \*pszXml)；

参数：

pszXml: 用于指定写入XML格式的信息；

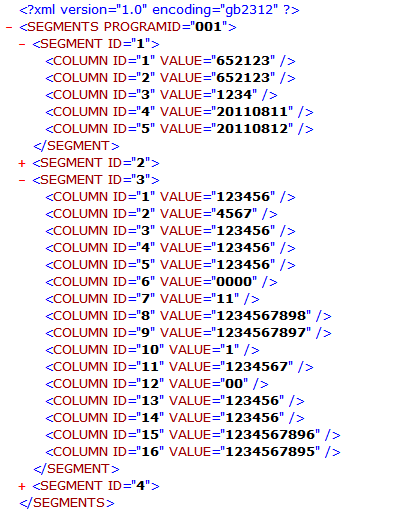
返回值：

如果写入成功，返回0，否者返回非零错误码。

备注：

无

XML文件格式见3.3.2



### 3.2.12 PatchCard

制卡的接口,完成补卡,包含卡面打印以及制卡.

int iPatchCard(

char \*CardDataXml,

char \*CardCoverDataXml,

char \*pszPrinterType，

char \*pszXZQH

)；

参数：

pszCardDataXml:卡数据XML格式的数据（格式见3.3.2）

pszCardCoverDataXml:卡面数据XML格式的数据格式见3.3.4）

pszPrinterType:打印机类型; （可为空，内部自己搜索连接上的打印机）

pszXZQH:卡所属人的行政区划，主要由于对应不同的卡面打印风格（目前是一个县一个卡面样式）。（格式见3.3.3）

返回值：

如果写入成功，返回0，否者返回非零错误码。

备注：

Xml的内存空间由外层应用分配.

批量的制卡由外层应用控制,本接口只完成一个人的制卡.

调用本接口之前必须调用iCardInit接口.

### 3.2.13 PrintCard

打印卡面接口,完成卡面打印。

int iPrintCard(

char \*pszPrinterType,

char \*CardCoverDataXml,

char \*pszXZQH

)；

参数：

pszCardCoverDataXml:卡面数据XML格式的数据（格式见3.3.4）

pszPrinterType:打印机类型（可为空，内部自己搜索连接上的打印机）

pszXZQH:卡所属人的行政区划，主要由于对应不同的卡面打印风格（目前是一个县一个卡面样式）.（格式见3.3.3）

返回值：

如果写入成功，返回0，否者返回非零错误码.

备注：

应用可调用此接口,只完成打印卡面的功能.

### 3.2.14 CreateCard

制卡接口,完成卡片的制作。

int iCreateCard(

pszCardDataXml,

)；

参数：

pszCardDataXml:卡数据XML格式的数据（格式见3.3.2）

pszCardDescritionXml: 用于描述卡数据格式的XML描述;

返回值：

如果写入成功，返回0，否者返回非零错误码。

备注：

制卡可通过卡设备以及内嵌卡打印机的读写模块.

调用本接口之前必须调用iCardInit接口.

### 3.2.15 FormatCard

将一张卡还原成原始卡.

**int iFormatCard();**

返回值：

如果写入成功，返回0，否者返回非零错误码。

### 3.2.16 CardIsEmpty

判断卡是否为空卡

**Int iCardIsEmpty()；**

参数：

无

返回值：

0 为空卡、 -1为写了数据的卡

### 3.2.17 err

根据错误码，返回详细错误信息。

**const char \*err()；**

参数：

无

返回值：

详细错误信息描述

备注：

无

### 3.2.18 EncryFile

加密文件，将加密的文件以“JM\_原文件名”形式保存在原文件夹中，以供卡商厂家制卡数据。

int iEncryFile(

char \*filepath,

)；

返回值：

如果写入成功，返回0，否者返回非零错误码。

### 3.2.19 GetPrinterList

获取打印机名字列表

int iGetPrinterList(

char \*PrinterXML,

)；

返回值：

如果成功，返回打印机个数，否者返回0。

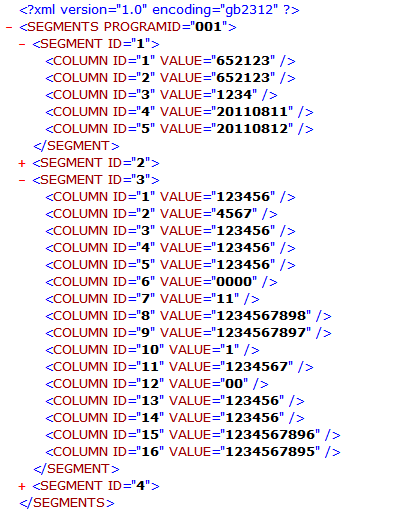
返回的XML文件格式：



* 1. 附件说明
     1. 卡初始化文件



* + 1. 卡读写文件



* + 1. 卡面打印风格文件



* + 1. 卡面数据文件

