

DB13

河北省地方标准

DB 13/T 2709—2018

法人和其他组织统一社会信用代码数据库 访问服务接口技术规范

2018-04-09 发布

2018-05-09 实施

河北省质量技术监督局 发布

前 言

本标准按照GB/T 1.1-2009给出的规则起草。

本标准由河北省质量技术监督局提出。

本标准由河北省服务标准化技术委员会（SAHB/TC 264）归口。

本标准起草单位：河北省标准化研究院、石家庄宇邦软件科技有限公司。

本标准起草人：阎占辉、董世涛、李虢峰、高龙、于潇、闫志强、李英君、孙明立、李会、孙家田。

法人和其他组织统一社会信用代码数据库 访问服务接口技术规范

1 范围

本标准规定了法人和其他组织统一社会信用代码数据库访问服务接口（以下简称统一代码接口）的术语、定义和缩略语、统一代码接口框架、统一代码接口要求、统一代码接口服务数据项和统一代码接口调用状态信息。

本标准适用于政府部门以接口方式进行的统一社会信用代码数据库信息共享应用。

2 规范性引用文件

下列文件对于本文件的应用是必不可少的。凡是注日期的引用文件，仅注日期的版本适用于本文件。凡是不注日期的引用文件，其最新版本（包括所有的修改单）适用于本文件。

GB/T 2260 中华人民共和国行政区划代码

GB/T 4754 国民经济行业分类

GB 32100 法人和其他组织统一社会信用代码编码规则

3 术语、定义和缩略语



3.1 术语和定义

全国组织机构代码统一社会信用代码数据服务中心

下列术语和定义适用于本文件。<https://www.cods.org.cn>

3.1.1

接口 interface

体现一个实体行为特征的具体名称的操作集。

3.1.2

请求 request

接口用户向接口发出的一次调用申请。

3.1.3

响应 response

接口向接口用户返回请求的操作结果。

3.1.4

包体 entity-body

数据传输中的实体主体部分。

3.2 缩略语

下列缩略语适用于本文件。

HTTPS 基于安全套接字的超文本传输协议 (Secure hypertext transfer protocol)

XML 可扩展标识语言 (Extensible markup language)

JSON JS对象标记 (JavaScript object notation)

4 统一代码接口框架

4.1 许可和调用

4.1.1 接口用户向接口提供方提交使用申请，用户申请信息包括：

- a) 机构统一社会信用代码；
- b) 机构名称；
- c) 用于实现应用接口功能的设备的 IP 地址；
- d) 申请使用期限；
- e) 申请使用数据项。

4.1.2 接口提供方收到申请后，生成许可信息反馈给接口用户，许可信息包括：

- a) 机构统一社会信用代码；
- b) 机构名称；
- c) 用于实现应用接口功能的设备的 IP 地址；
- d) 申请使用期限；
- e) 用户账号；
- f) 用户密码；
- g) 接口地址。

全国组织机构代码统一社会信用代码数据服务中心
<https://www.cods.org.cn>

4.1.3 接口用户在调用接口时，应提供被授予的许可信息。

4.1.4 接口在接收到请求后，校验接口用户许可信息中用于实现应用接口功能的设备的 IP 地址和申请使用期限的有效性、用户账号和用户密码的合法性，检验通过后，执行请求，返回响应内容。

4.2 应用模型

统一代码接口应用模型见图1。

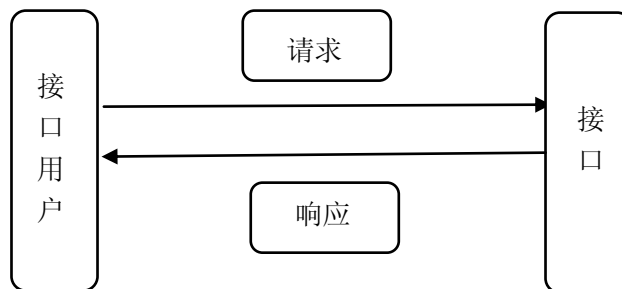


图1 统一代码接口应用模型

5 统一代码接口要求

5.1 接口定义

统一代码接口定义见表1。

表 1 统一代码接口定义

定义	说明
String Tydm_sh_service (string licence, string cond_information, string repl_packagetype)	<ol style="list-style-type: none"> 1. Tydm_sh_service 为约定的接口名称。 2. license 为许可参数(见 5.3)。 3. cond_information 为条件参数(见 5.4)。 4. repl_packagetype 为返回结果数据包格式参数(见 5.5)。 5. 接口返回结果见 5.6。

5.2 基本约定

5.2.1 传输协议

采用HTTPS作为传输协议。

5.2.2 地址格式

接口地址由接口提供方提供，格式为：https://xxx.xxx.xxx.xxx/services_path/service_name。

其中：xxx.xxx.xxx.xxx表示接口所在的服务器的有效地址；services_path表示接口的发布路径，由接口提供方命名；service_name表示约定的接口名称，由接口提供方命名。

<https://www.cods.org.cn>

5.2.3 安全性

接口用户以接口提供方授权的许可为凭据调用接口。

5.2.4 调用方法

统一代码接口调用方法实例参见附录A。

5.3 许可参数

5.3.1 内容组成

许可参数由一组用户信息组成。

5.3.2 包体标记及含义

许可参数包体标记及含义说明见表2。

表 2 许可参数包体标记及含义

信息类别	标记	字符类型及长度	含义	说明
用户信息	tydm	nvarchar(18)	机构统一社会信用代码	---
	jgmc	nvarchar(400)	机构名称	---
	appkey	nvarchar(8)	用户账号	---
	appsecret	nvarchar(10)	用户密码	---
	ip	nvarchar(15)	被授权应用接口的设备的 IP 地址	---
	startdate	nvarchar(10)	使用期限开始时间	授权起始日期
	enddate	nvarchar(10)	使用期限结束时间	授权终止日期

5.3.3 包体格式说明

许可参数包体采用JSON格式。
许可参数放置在标记 {...} 之间，见示例1。

示例1:

```
{
  "tydm": "xxxxxxxxxxxxxxxxxxx",
  "jgmc": "xxx",
  "appkey": "xxx",
  "appsecret": "xxx",
  "ip": "xxx",
  "startdate": "xxx",
  "enddate": "xxx"
}
```



全国组织机构代码统一社会信用代码数据服务中心
<https://www.cods.org.cn>

5.4 条件参数

5.4.1 内容组成

条件参数由条件逻辑关系、条件数据项集、返回结果数据项名称组成。见图2。

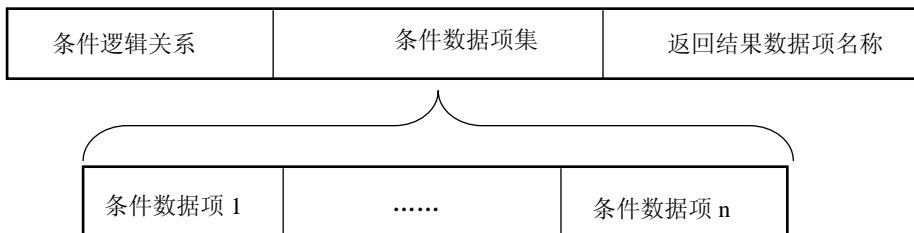


图 2 条件参数

5.4.2 包体格式说明

条件参数包体采用JSON格式。

条件参数放置在标记 {"exp": "xxx", "fields": [{"field": "xxx", "operation": "xxx", "value": "xxx"}, ..., {"field": "xxx", "operation": "xxx", "value": "xxx"}], "result": "xxx"} 之间。

exp关键字为条件逻辑关系。其中与、或分别用*、+标识；为空时，默认为与；在逻辑关系中，第一个条件数据项用[0]表示，第二个用[1]表示，以此类推，第n个用[n-1]表示，各条件之间可以用圆括号运算符（）组合。

fields关键字为条件数据项集。其中field关键字为条件数据项；operation为条件数据项的关系运算符，其表示及含义见表3；value为条件数据项的值。

result关键字为返回结果中用户要求的数据项名称，各数据项间以“，”分隔。

数据项标识应符合本标准第6章的规定。

表 3 关系运算符表示及含义

关系运算符	含义	说明
=	等于	字符型按字符编码顺序进行比较 数值型应转换成数值进行比较 日期型应转换成日期进行比较 日期时间型应转换成时间进行比较
<	小于	
>	大于	
<=	小于或等于	
>=	大于或等于	
<>	不等于	
like	模糊匹配	仅可应用于字符型的数据项，从字符串任意位置进行匹配
left like	左匹配	仅可应用于字符型的数据项，从字符串起始端匹配
right like	右匹配	仅可应用于字符型的数据项，从字符串尾端匹配
is null	空值	---
is not null	非空	---

条件参数包体格式见示例2。

示例2:

```
{
  "exp": "[0]*([1]*[2])",
  "fields": [
    {
      "field": "xzqh",
      "operation": "=",
      "value": "xxx"
    },
    {
      "field": "jgmc",
      "operation": "like",
      "value": "xxx"
    },
    {
      "field": "jgmc",
      "operation": "like",
      "value": "xxx"
    }
  ]
}
```

```

    }
  ]
  "result": "tydm, jgmc, xzqh"
}

```

5.5 返回结果数据包格式参数

接口返回结果数据包格式参数由接口用户根据需要设定。该参数取值可为xml或json，不能为空。

5.6 返回结果

5.6.1 内容组成

返回结果包括接口调用状态信息、结果数据和结果集记录个数组成。接口调用失败时，只返回接口调用状态信息。见图3。

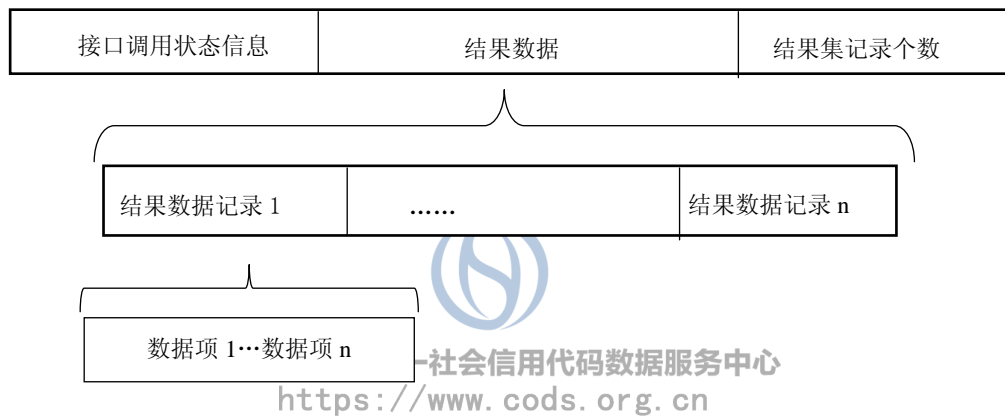


图3 返回结果

5.6.2 包体格式说明

5.6.2.1 XML 格式

XML格式采用标准DATAPACKET 2.0版本。

返回结果中的接口调用状态和结果集记录个数信息放置在标记<RESPDATA>...</RESPDATA>之间，关键字status为调用状态代码，msg为接口调用状态描述，totalcount为结果集记录个数。接口调用状态代码和描述应符合本标准第7章的规定。

结果数据置于标记<ROWDATA>...</ROWDATA>之间。结果数据不同记录间用<ROW.../>...<ROW.../>分隔。数据项标识应符合本标准第6章的规定。

返回结果包体XML格式见示例3。

示例3:

```

<?xml version="1.0" encoding="UTF-8"?>
<DATAPACKET Version="2.0">
  <RESPDATA>
    <RESP status="0" msg="调用成功" totalcount="2" />
  </RESPDATA>
  <ROWDATA>
    <ROW tydm="XXXXXXXXXXXXXXXXXXXX" jgmc="河北省 XXX 股份有限公司" xzqh="XXXXXX"/>
    <ROW tydm="XXXXXXXXXXXXXXXXXXXX" jgmc="石家庄市 XXX 股份有限公司" xzqh="XXXXXX"/>
  </ROWDATA>
</DATAPACKET>

```

```
</ROWDATA>
</DATA PACKET>
```

5.6.2.2 JSON 格式

返回结果信息放置在标记{"respdata":{"status":"xxx","msg":"xxx","totalcount":"xxx"}, "rowdata":[{"xxx}, ..., {xxx}]}之间。

在respdata标签内，关键字status为接口调用状态代码，msg为接口调用状态描述，totalcount为结果集记录个数。接口调用状态代码和描述应符合本标准第7章的规定。

在rowdata标签内，每条结果数据记录放置在标记{...}之间。数据记录中的数据项标识应符合本标准第6章的规定。

返回结果包体JSON格式见示例4。

示例 4:

```
{
  "respdata": {
    "status": "0",
    "msg": "调用成功",
    "totalcount": "2"
  },
  "rowdata": [
    {
      "tydm": "XXXXXXXXXXXXXXXXXX",
      "jgmc": "河北省 XXX 股份有限公司",
      "xzqh": "XXXXXX"
    },
    {
      "tydm": "XXXXXXXXXXXXXXXXXX",
      "jgmc": "石家庄市 XXX 股份有限公司",
      "xzqh": "XXXXXX"
    }
  ]
}
```

6 统一代码接口服务数据项

接口服务提供的数据项见表4。

表 4 接口服务数据项

数据项标识	含义	数据类型及长度	说明
tydm	统一社会信用代码	nvarchar(18)	符合 GB 32100
jgmc	机构名称	nvarchar(400)	---
fddbr	法定代表人或负责人	nvarchar(255)	---
xzqh	行政区划代码	nvarchar(6)	符合 GB/T 2260
jgdz	机构地址	nvarchar(500)	---
jyfw	业务或经营范围	nvarchar(3000)	---
sshy	所属经济行业代码	nvarchar(5)	符合 GB/T 4754

表 4 接口服务数据项 (续)

数据项标识	含义	数据类型及长度	说明
lxdh	联系电话	nvarchar(50)	---
clrq	登记成立日期	Datetime	---
djzt	登记状态	nvarchar(2)	0: 新办, 1: 变更, 2: 注销
gxrq	最后更新日期	Datetime	---

7 统一代码接口调用状态信息

接口调用状态信息包括接口状态代码及接口状态描述, 详见表5。

表 5 接口调用状态代码及接口调用状态描述

接口状态代码	接口状态描述
0	调用成功
1	调用失败
-100	被授权应用接口的设备 IP 地址错误
-130	条件参数错误
-160	返回结果数据包格式参数错误
-200	数据库错误
-300	用户账号错误
-400	用户密码错误

附录 A

(资料性附录)

统一代码接口调用方法实例

A.1 实例说明

在本实例中,设定统一代码接口地址为:https://www.hbjgdm.gov.cn:8082/dso/Tydm_sh_service;接口用户的机构统一社会信用代码为:111300004017053009,机构名称为:河北省标准化研究院;通过接口查询的数据条件是:机构名称中包含“标准”且包含“研究院”的机构;返回结果包体是:json格式,结果数据项包括:统一代码、机构名称、法定代表人或负责人、机构地址、经营范围、联系电话、成立日期、所属经济行业代码、行政区划代码、登记状态、最后更新日期。

A.2 实例程序代码

本实例以 java 语言实现,程序代码如下:

```
package testjgdm; //包名 testjgdm
import java.net.URLEncoder;
import java.util.HashMap;
import java.util.Map;
import org.apache.http.HttpResponse;
import org.apache.http.util.EntityUtils;
public class jgdm { //定义类 jgdm
public static void main(String[] args) //调用统一代码接口获取符合条件数据的函数
{
String strkey="标准"; //查询关键字1: //www.cods.org.cn
String strkey2="研究院"; //查询关键字 2
try {
    strkey = URLEncoder.encode(strkey, "utf-8"); //将查询关键字转换为 utf-8 编码
    strkey2 = URLEncoder.encode(strkey2, "utf-8");
} catch (Exception e1) {
    e1.printStackTrace();
}
String host="https://www.hbjgdm.gov.cn:8082"; //接口地址
String service_path = "/dso/"; //接口路径
String serivce_name = "Tydm_sh_service"; //接口名称
String method = "POST"; //提交方式
String appkey="NyrcP"; //许可参数中的 appkey
String appsecret="jttLM7wu"; //许可参数中的 appsecret
String tydm="111300004017053009"; //许可参数中的 tydm
String jgmc="河北省标准化研究院"; //许可参数中的 jgmc
String ip ="121.28.48.222"; //许可参数中的 ip
String startdate ="2016-02-02"; //许可参数中的 startdate
String enddate ="2017-08-08"; //许可参数中的 enddate
Map headers = new HashMap();
Map querys = new HashMap();
//构建许可参数 licence
querys.put("licence", "{\tydm\":" + tydm + "\",\tjgmc\":" + jgmc + "\",\tappkey\":" + appkey + "\",\tappsecret\":" + appsecret + "\",\tip\":" + ip + "\",\tstartdate\":" + startdate + "\",\tenddate\":" + enddate + "\"}");
//构建返回结果格式参数 repl_packagetype
querys.put("repl_packagetype", "json"); //返回结果数据包为 json 格式
```

```
//构建条件参数 cond_information
    queries.put("cond_information", "{\nexp\": \"[0]*[1]\", \nresult\": \"tydm, jgmc,\nfdbr, jgdz, jyfw, lxdh, clrq, sshy, xzqh, djzt, gxrq\", \nfields\": [\nfield\": \"jgmc\", \noperation\":\nlike\", \nvalue\": \"\"+strkey+\"\", {\nfield\": \"jgmc\", \noperation\": \"like\", \nvalue\":\n\"\"+strkey2+\"\"}]]");
try { //向接口提交请求
    HttpResponse response = HttpUtils.doPost(host+service_path, service_name, method, headers,
queries, "");
    //获取反馈的数据包, 并在控制窗口中打印出来
    System.out.println(EntityUtils.toString(response.getEntity()));
} catch (Exception e) {
    e.printStackTrace();
}
}
```
