

电子凭证会计数据标准深化试点个性化工具包  
使用说明文档  
( V1.0 )

北京四方启点科技有限公司  
2023 年 5 月

# 前 言

本说明文档的目标读者是电子凭证会计数据标准深化试点接收端试点单位的技术人员，用于了解通过使用本工具包来对符合电子凭证会计数据标准的电子凭证进行验签、解析、转换等操作。接收端试点单位的会计信息系统可通过集成本个性化工具包来降低深化试点技术难度和实施成本。

作为深化试点服务保障单位，如您在使用本工具包过程中有相关疑问，请您联系：

联系人：公务之家工具包技术支持

联系电话：（010）68118224 转 627

邮箱：[support@gwhome.com.cn](mailto:support@gwhome.com.cn)

## 文档修订记录

[illegible]

## 一、权利与义务声明

北京四方启点科技有限公司是公务之家电子凭证网上报销服务平台的建设和运营方。

该工具包由北京四方启点科技有限公司免费提供给电子凭证会计数据标准试点专班。有需要的试点单位可根据自身情况选择使用。选择使用该工具包的单位视同为接受以下权利和义务：

1、选择使用该工具包的单位或个人被允许将该工具包集成到自主的软件系统或产品中，用于自用或商业用途。

2、选择使用该工具包的单位或个人不被允许依托该工具包申请专利或软件著作权等知识产权。

3、工具包发布方公开提供工具包的哈希校验值，选择使用该工具包的单位或个人需要自行检测所获取工具包的完整性。

4、选择使用该工具包的单位或个人应自行保障该工具包被集成到自主的软件系统或产品中的安全性，工具包发布方不负责识别由于可能未正确使用该工具包而导致的系统安全风险。

## 二、工具包使用方法

本工具包的主要功能是通过 API 从电子凭证文件中解析提取结构化数据，组装为 JSON 对象，并对电子凭证中的数字签名进行验证。

### 1、依赖包说明

工具包提供 API 的 jar 包（gwhome-xbrl-1.0.jar、

gwhome-base-1.0.jar)。

2、接口 1：电子凭证解析及验签接口

序号	API	描述
1	static JSONObject getXbrlJsonWithPath(String path, String voucherType)	从 OFD/ZIP/PDF 等格式的电 子凭证中解析出结构化数据 文件，并验签

入参说明：

序号	名称	类型	必须	描述
1	path	String	是	电子凭证文件路径
2	voucherType	String	是	电子凭证类型(见三、接收方接 收到的电子凭证类型枚举)

出参说明：

序号	名称	类型	必须	描述
1	Suc	Boolean	是	是否解析成功
2	Verified	Boolean	是	是否数字签名验签成功
3	RootVerified	Boolean	是	是否根证书验签成功
4	XbrlJson	JSONObject	是	XbrlJson 对象
5	SignatureInfo	JSONObject	否	签名证书信息
6	ErrorMsg	String	是	解析失败原因 (如：票据类型不存在)
7	Remark	String	否	备注 (如：未验签)

请注意：

(1) 由于财政电子票据暂不做验签，因此

- 若 Verified 为 true，表示电子凭证验签通过；
- 若 Verified 为 false，且 Remark 不为【未验签】，表示电子凭证验签不通过；
- Verified 为 false，且 Remark 为【未验签】，表示该凭证未做验签（如财政电子票据类型）。

(2) 接口调用成功失败的首要判断标识是：Suc 节点。

接口调用代码示例：

```
package com.gwhome.xbrl_test;

import com.alibaba.fastjson.JSONObject;
import com.gwhome.xbrl.XbrlUtils;

public class Test {

    public static void main(String[] args) {
        try {
            // rai_issuer
            String filePath = "E:\\\\售票样例.ofd";
            JSONObject json =
XbrlUtils.getXbrlJsonWithPath(filePath, "rai_issuer");
            System.out.println("result==>" + json);
        } catch (Exception e) {
            e.printStackTrace();
        }
    }
}
```

输出结果示例：

```
{
  "Suc": true,
  "Verified": true,
  "SignatureInfo": {
    "IssuerDN": "SRCA_SM2",
    "NotBefore": "2022-03-10",
    "NotAfter": "2027-03-10",
    "SubjectDN": "电子发票（铁路电子客票）"
  },
  "ErrorMsg": "",
  "XbrlJson": {
    "TypeOfVoucher": "电子发票（铁路电子客票）",
    "ElectronicInvoiceRailwayETicketNumber":
"22119130671000000049",
    "DateOfIssue": "2022-03-17",
    "IssueParty": "",
    "IssuePartyCode": "",
    "TypeOfBusiness": "售",
    "DepartureStation": "南京",
    "PhonicsOfDepartureStation": "Nanjing",
    "DestinationStation": "上海虹桥",
    "PhonicsOfDestinationStation": "Shanghaihongqiao",
    "TrainNumber": "G7101",
    "TravelDate": "2022-03-01",
    "DepartureTime": "06:23",
    "TypeOfRailwayTicket": "",
    "AirConditioningCharacteristics": "新空调",
    "SeatLevel": "软卧",
    "Carriage": "01 车",
    "Seat": "05C 号",
    "DiscountMark": "",
    "Fare": "181.00",
    "AmountRefunded": "",
    "FareOfOriginalRailwayTicket": "",
    "DepartureStationOfOriginalRailwayTicket": "",
    "DestinationStationOfOriginalRailwayTicket": "",
    "ETicketNumber": "306712A086012090014302022",
    "IdNumber": "1427241988****1627",
    "Name": "梁某某",
    "TotalAmountExcludingTax": "166.05",
    "TaxRate": "0.09",
    "TaxAmount": "14.95",
    "NameOfPurchaser": "铁路客票电子发票测试单位",
    "UnifiedSocialCreditCodeOfPurchaser": "91110001110AE35858",
```

```
        "AddressPhoneNumberOfPurchaser": "",
        "DepositBankAndAccountNumberOfPurchaser": "",
        "NumberOfOriginalInvoice": "22119130671000000048",
        "Remarks": ""
    },
    "RootVerified": true
}
```

### 3、接口 2：XML 数据转换为 JSON 数据接口

序号	API	描述
1	<code>public static JSONObject xmlToJson(String path)</code>	将 XML 格式的电子凭证 转换为 JSON 对象

入参说明：

序号	名称	类型	必须	描述
1	path	String	是	XML 文件路径

返回结果：

序号	类型	描述
1	JSONObject	JSON 格式结构化数据

接口调用代码示例：

```
package com.gwhome.xbrl_test;

import com.alibaba.fastjson.JSONObject;
import com.gwhome.xbrl.XbrlUtils;

public class Test {

    public static void main(String[] args) {
        try {
            // 数电票
            String filePath = "E:\\数电票样例.xml";
```



```

        JSONObject json = XbrlUtils.xmlToJson(filePath);
        System.out.println("result==>" + json);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

```

输出结果示例：

```

{
  "Header": [{
    "EIId": "23442000000000130159",
    "EInvoiceTag": "SWEI4400",
    "Version": "0.2",
    "InherentLabel": [{
      "InIssuType": [{
        "LabelCode": "Y",
        "LabelName": "是否蓝字发票标志"
      }],
      "EInvoiceType": [{
        "LabelCode": "01",
        "LabelName": "电子发票"
      }],
      "GeneralOrSpecialVAT": [{
        "LabelCode": "01",
        "LabelName": "增值税专用发票"
      }],
      "TaxpayerType": [{
        "LabelCode": "1",
        "LabelName": "一般纳税人"
      }]
    }],
  "UndefinedLabel": [{
    "Label": [{
      "LabelType": "企业行业性质类标签",
      "LabelCode": "01,20",
      "LabelName": "成品油生产企业,生产型出口企业"
    }, {
      "LabelType": "发票开具方式标签",
      "LabelCode": "1",

```

```
        "LabelName": "网页开票"
    }]
  }],
  "EInvoiceData": [{
    "SellerInformation": [{
      "SellerIdNum": "91440600065160777Y",
      "SellerName": "箭牌家居集团股份有限公司",
      "SellerAddr": "佛山市三水区南山镇康裕三路1号1座(住所申报)",
      "SellerTelNum": "13702921726",
      "SellerBankName": "中国工商银行股份有限公司佛山南海同乐支行",
      "SellerBankAccNum": "2432434"
    }],
    "BuyerInformation": [{
      "BuyerIdNum": "914406065608413420",
      "BuyerName": "佛山市顺德区尊品无纺布制品有限公司",
      "BuyerAddr": "",
      "BuyerTelNum": "",
      "BuyerBankName": "",
      "BuyerBankAccNum": "",
      "BuyerHandlingName": ""
    }],
    "BasicInformation": [{
      "TotalAmWithoutTax": "2.00",
      "TotalTaxAm": "0.12",
      "TotalTax-includedAmount": "2.12",
      "TotalTax-includedAmountInChinese": "贰圆壹角贰分",
      "Drawer": "谢岳荣",
      "RequestTime": "2023-02-23"
    }],
    "IssuItemInformation": [{
      "ItemName": "*人力资源服务*劳务",
      "SpecMod": "",
      "MeaUnits": "",
      "Quantity": "2",
      "UnPrice": "1",
      "Amount": "2.00",
      "TaxRate": "0.060000",
      "ComTaxAm": "0.120000",
      "TotaltaxIncludedAmount": "2.120000",
      "TaxClassificationCode": "3040803010000000000"
    }],
  }],
```

```
        "SpecificInformation": "",
        "AdditionalInformation": [{
            "Remark": "备注说明"
        }]
    }],
    "SellerAuthentication": [{
        "AuthenticationMethods": "02"
    }],
    "TaxSupervisionInfo": [{
        "InvoiceNumber": "234420000000000130159",
        "IssueTime": "2023-02-23 10:08:05",
        "TaxBureauCode": "14400000000",
        "TaxBureauName": "国家税务总局广东省税务局"
    }],
    "TaxBureauSignature": [{
        "Reference": "",
        "SignatureAlgorithm": "1.2.156.10197.1.501",
        "SignatureFormat": "DETACH",
        "SignatureTime": "2023-02-23 10:08:55",
        "SignatureValue": "",
        "KeyInfo": ""
    }]
}
```

4、接口 3: JSON 格式结构化数据转换为 XBRL 实例接口

序号	API	描述
3	<code>public static String json2Xbrl(String jsonstr, String configID);</code>	将 JSON 对象格式的结构化数据按类型（configID）转换为 XBRL 实例文件

入参说明：

序号	名称	类型	必须	描述
1	jsonstr	String	是	待转换的 json 结构化数据字符串
2	configID	String	是	目标 XBRL 实例文档类型



```

\R0009\","\r\n"
+ "          \\"PostingDate\\": \\"2022-06-30\\",\r\n"
+ "          \\"AccountingPeriod\\": \\"2022-06\\",\r\n"
+ "          \\"InformationOfDebitAndCreditEntryTuple\\":
[\r\n"
+ "              {\r\n"
+ "                  \\"DebitOrCredit\\": \\"借方\\",\r\n"
+ "                  \\"NameOfGeneralLedgerSubject\\": \\"
原材料\\",\r\n"
+ "                  \\"RecordedAmount\\": \\"9000.00\\",\r\n"
+ "                  },\r\n"
+ "                  {\r\n"
+ "                      \\"DebitOrCredit\\": \\"借方\\",\r\n"
+ "                      \\"NameOfGeneralLedgerSubject\\": \\"
应交税费\\",\r\n"
+ "                      \\"NameOfSubsidiaryLedgerSubject\\":
\\"应交税费-进项税额\\",\r\n"
+ "                      \\"RecordedAmount\\": \\"1170.00\\",\r\n"
+ "                      },\r\n"
+ "                      {\r\n"
+ "                          \\"DebitOrCredit\\": \\"贷方\\",\r\n"
+ "                          \\"NameOfGeneralLedgerSubject\\": \\"
银行存款\\",\r\n"
+ "                          \\"RecordedAmount\\": \\"10170\\",\r\n"
+ "                          },\r\n"
+ "                      ]\r\n"
+ "                  }\r\n"
+ "              ],\r\n"
+ "          "
\\"WhetherPreTaxDeductionOfIncomeTaxHasBeenCompleted\\": \\"false\\",\r\n"
+ "          \\"BeginningOfExpenditurePeriodUnderAccrualBasis\\":
\\"2022-06\\",\r\n"
+ "          \\"WhetherEinvoiceUsageHasBeenConfirmed\\":
\\"true\\",\r\n"
+ "          \\"UsageConfirmation\\": \\"xxxx\\",\r\n"
+ "          \\"WhetherInputVatHasBeenTransferredOut\\":
\\"true\\",\r\n"
+ "          \\"AmountOfTransferredOutInputVat\\":
\\"1170.00\\",\r\n"
+ "          \\"PeriodOfUsageConfirmation\\": \\"2022-06\\",\r\n"
+ "          \\"BeginningOfPreTaxDeductionYearOfIncomeTax\\":
\\"2022\\",\r\n"
+ "          \\"EndOfPreTaxDeductionYearOfIncomeTax\\":
\\"2022\\",\r\n"

```

```

        + "    \"EndOfExpenditurePeriodUnderAccrualBasis\": \"2022-12\", \r\n"
        + "    \"SellerName\": \"中山市奕博照明有限公司\", \r\n"
        + "    \"SellerIdNum\": \"91442000MA52PFXD1J\", \r\n"
        + "    \"TotalAmWithoutTax\": \"9000\", \r\n"
        + "    \"RequestTime\": \"2022-06-30\", \r\n"
        + "    \"TotalTaxAm\": \"1170\", \r\n"
        + "    \"TotalTax-includedAmount\": \"10170\" \r\n"
        + "}"
        String xbrlstr=XbrlUtils.json2Xbrl(jsonstr,
"einv_spcl_receiver");
        System.out.println("result==>" + xbrlstr);
    } catch (Exception e) {
        e.printStackTrace();
    }
}
}

```

输出结果示例:

```

<xbrl
  xmlns="http://www.xbrl.org/2003/instance"
  xmlns:einv="http://xbrl.mof.gov.cn/taxonomy/2022-09-30/einv"
  xmlns:link="http://www.xbrl.org/2003/linkbase"
  xmlns:iso4217="http://www.xbrl.org/2003/iso4217"
  xmlns:xbrli="http://www.xbrl.org/2003/instance"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:xlink="http://www.w3.org/1999/xlink"
  xsi:schemaLocation="http://www.xbrl.org/2003/linkbase
http://www.xbrl.org/2003/xbrl-linkbase-2003-12-31.xsd">
  <link:schemaRef xlink:type="simple"
xlink:href="einv_entry_point_2022-09-30.xsd"/>
  <context id="c1">
    <entity>
      <identifier
scheme="http://xbrl.mof.gov.cn">91440101MA5CLQ731R</identifier>
    </entity>
    <period>
      <instant>2022-06-30</instant>
    </period>
  </context>
  <unit id="u1">

```

```

        <measure>iso4217:CNY</measure>
    </unit>
    <einv:InvoiceNumber
contextRef="c1">22442000000921291350</einv:InvoiceNumber>
    <einv:ContractNumber contextRef="c1">R9988888</einv:ContractNumber>
    <einv:MatchingStateBetweenBusinessDocumentsAndVatEinvoice
contextRef="c1">已匹配
</einv:MatchingStateBetweenBusinessDocumentsAndVatEinvoice>
    <einv:WhetherEinvoiceIsRedEinvoice
contextRef="c1">false</einv:WhetherEinvoiceIsRedEinvoice>
    <einv:WhetherEinvoiceHasBeenBooked
contextRef="c1">true</einv:WhetherEinvoiceHasBeenBooked>
    <einv:WhetherEinvoiceHasBeenChecked
contextRef="c1">true</einv:WhetherEinvoiceHasBeenChecked>
    <einv:WhetherEinvoiceHasBeenPaid
contextRef="c1">true</einv:WhetherEinvoiceHasBeenPaid>
    <einv:NumberOfBankElectronicReceipt
contextRef="c1">6FE0-4F2C-88AA</einv:NumberOfBankElectronicReceipt>
    <einv:UnifiedSocialCreditCodeOfAccountingEntity
contextRef="c1">91440101MA5CLQ731R</einv:UnifiedSocialCreditCodeOfAccountin
gEntity>
    <einv:NameOfAccountingEntity contextRef="c1">广州市兆丰商贸有限公司
</einv:NameOfAccountingEntity>
    <einv:InformationOfAccountingDocumentsTuple>
        <einv:NumberOfAccountingDocuments
contextRef="c1">R0009</einv:NumberOfAccountingDocuments>
        <einv:PostingDate contextRef="c1">2022-06-30</einv:PostingDate>
        <einv:AccountingPeriod
contextRef="c1">2022-06</einv:AccountingPeriod>
        <einv:InformationOfDebitAndCreditEntryTuple>
            <einv:DebitOrCredit contextRef="c1">借方</einv:DebitOrCredit>
            <einv:NameOfGeneralLedgerSubject contextRef="c1">原材料
</einv:NameOfGeneralLedgerSubject>
            <einv:RecordedAmount contextRef="c1" unitRef="u1"
decimals="2">9000.00</einv:RecordedAmount>
        </einv:InformationOfDebitAndCreditEntryTuple>
        <einv:InformationOfDebitAndCreditEntryTuple>
            <einv:DebitOrCredit contextRef="c1">借方</einv:DebitOrCredit>
            <einv:NameOfGeneralLedgerSubject contextRef="c1">应交税费
</einv:NameOfGeneralLedgerSubject>
            <einv:NameOfSubsidiaryLedgerSubject contextRef="c1">应交税费-进项
税额</einv:NameOfSubsidiaryLedgerSubject>
            <einv:RecordedAmount contextRef="c1" unitRef="u1"
decimals="2">1170.00</einv:RecordedAmount>

```

```

        </einv:InformationOfDebitAndCreditEntryTuple>
        <einv:InformationOfDebitAndCreditEntryTuple>
            <einv:DebitOrCredit contextRef="c1">贷方</einv:DebitOrCredit>
            <einv:NameOfGeneralLedgerSubject contextRef="c1">银行存款
</einv:NameOfGeneralLedgerSubject>
            <einv:RecordedAmount contextRef="c1" unitRef="u1"
decimals="2">10170.00</einv:RecordedAmount>
        </einv:InformationOfDebitAndCreditEntryTuple>
    </einv:InformationOfAccountingDocumentsTuple>
    <einv:WhetherPreTaxDeductionOfIncomeTaxHasBeenCompleted
contextRef="c1">false</einv:WhetherPreTaxDeductionOfIncomeTaxHasBeenComple
ted>
    <einv:BeginningOfExpenditurePeriodUnderAccrualBasis
contextRef="c1">2022-06</einv:BeginningOfExpenditurePeriodUnderAccrualBasis
>
    <einv:WhetherEinvoiceUsageHasBeenConfirmed
contextRef="c1">true</einv:WhetherEinvoiceUsageHasBeenConfirmed>
    <einv:UsageConfirmation contextRef="c1">xxxxx</einv:UsageConfirmation>
    <einv:WhetherInputVatHasBeenTransferredOut
contextRef="c1">true</einv:WhetherInputVatHasBeenTransferredOut>
    <einv:AmountOfTransferredOutInputVat contextRef="c1" unitRef="u1"
decimals="2">1170.00</einv:AmountOfTransferredOutInputVat>
    <einv:PeriodOfUsageConfirmation
contextRef="c1">2022-06</einv:PeriodOfUsageConfirmation>
    <einv:BeginningOfPreTaxDeductionYearOfIncomeTax
contextRef="c1">2022</einv:BeginningOfPreTaxDeductionYearOfIncomeTax>
    <einv:EndOfPreTaxDeductionYearOfIncomeTax
contextRef="c1">2022</einv:EndOfPreTaxDeductionYearOfIncomeTax>
    <einv:EndOfExpenditurePeriodUnderAccrualBasis
contextRef="c1">2022-12</einv:EndOfExpenditurePeriodUnderAccrualBasis>
    <einv:SellerName contextRef="c1">中山市奕博照明有限公司</einv:SellerName>
    <einv:SellerIdNum contextRef="c1">91442000MA52PFXD1J</einv:SellerIdNum>
    <einv:TotalAmWithoutTax contextRef="c1" unitRef="u1"
decimals="2">9000.00</einv:TotalAmWithoutTax>
    <einv:RequestTime contextRef="c1">2022-06-30</einv:RequestTime>
    <einv:TotalTaxAm contextRef="c1" unitRef="u1"
decimals="2">1170.00</einv:TotalTaxAm>
    <einv:TotalTax-includedAmount contextRef="c1" unitRef="u1"
decimals="2">10170.00</einv:TotalTax-includedAmount>
</xbrl>

```



### 三、接收方接收到的电子凭证类型枚举

下表列示了工具包当前版本支持的接收方接收到的电子凭证类型，及业务系统调用工具包接口时对应传入的凭证类型编号参数。

需要注意的是，随着电子凭证会计数据标准深化试点的推进，电子凭证类型以及开具地区或单位将逐步增加，工具包的迭代版本中将逐步增加新类型电子凭证的支持。

电子凭证类型	地区/单位	文件格式	接口调用参数 凭证类型编号
数字化电子发票	全国	xml	dzfp
财政电子票据	中央单位	pdf	efi
增值税 电子普通发票	仅国家电网开具	zip	inv_ord_issuer
增值税 电子专用发票	仅国家电网开具	ofd	inv_spcl_issuer
铁路电子客票	中国国家铁路集团有限公司	ofd	rai_issuer
航空运输电子客票行程单	航空公司/航空代理公司	ofd	atr_issuer
电子非税收入一般缴款书	中央单位	pdf	ntrev_gpm_issuer
银行电子回单	工行银行	ofd	bker_issuer
	光大银行	ofd	bker_issuer
	浙商银行	ofd	bker_issuer
	盛京银行	ofd	bker_issuer
银行电子对账单	工行银行	ofd	bkrs
	光大银行	ofd	bkrs
	浙商银行	ofd	bkrs
	盛京银行	ofd	bkrs

### 四、JSON 结构化数据转换 XBRL 实例文件类型枚举

下表列示了工具包当前版本支持的 JSON 结构化数据转换为 XBRL 实例文档的凭证文件类型，及业务系统调用工具包接口时

对应传入的文件类型参数。

需要注意的是，随着电子凭证会计数据标准深化试点的推进，电子凭证类型将逐步增加，工具包的迭代版本中将逐步增加新类型电子凭证的支持。

序号	文件类型名称	接口调用参数（configID）
1	增值税电子普通发票（开具方）	inv_ord_issuer
2	增值税电子普通发票（接收方）	inv_ord_receiver
3	增值税电子专用发票（开具方）	inv_spcl_issuer
4	增值税电子专用发票（接收方）	inv_spcl_receiver
5	收费公路通行费增值税电子普通发票（开具方）	inv_tlf_issuer
6	收费公路通行费增值税电子普通发票（接收方）	inv_tlf_receiver
7	电子非税收入一般缴款书（开具方）	ntrev_gpm_issuer
8	电子非税收入一般缴款书（接收方）	ntrev_gpm_receiver
9	电子发票（铁路电子客票）（开具方）	rai_issuer
10	电子发票（铁路电子客票）（接收方）	rai_receiver
11	电子发票（航空运输电子客票行程单）（开具方）	atr_issuer
12	电子发票（航空运输电子客票行程单）（接收方）	atr_receiver
13	银行电子回单（开具方）	bker_issuer
14	银行电子回单（接收方）	bker_receiver
15	银行电子对账单（开具方）	bkrs
16	全面数字化的电子发票-普通发票（接收方）	einv_ord_receiver
17	全面数字化的电子发票-增值税专用发票（接收方）	einv_spcl_receiver
18	财政电子票据（接收方）	efi