ICS 03.060
CCS A 11

JR

中 华 人 民 共 和 国 金 融 行 业 标 准

JR/T XXXXX—XXXX

区域性股权市场区块链跨链技术规范

Specification for regional equity markets cross-chain technology

（送审稿）

XXXX-XX-XX 发布　　XXXX-XX-XX 实施

中国证券监督管理委员会　　发 布

目　　次

前　言

本文件按照GB/T　1.1—2020《标准化工作导则　第1部分：标准化文件的结构和起草规则》的规定起草。

本文件由全国金融标准化技术委员会证券分技术委员会（SAC/TC　180/SC4）提出。

本文件由全国金融标准化技术委员会（SAC/TC　180）归口。

本文件起草单位：中国证券监督管理委员会科技监管局、中国证券监督管理委员会市场二部、中证信息技术服务有限责任公司、深圳证券通信有限公司、北京股权交易中心有限公司、上海股权托管交易中心股份有限公司、江苏股权交易中心有限责任公司、浙江省股权交易中心有限公司、深圳前海股权交易中心有限公司、上海边界智能科技有限公司、中诚区块链研究院（南京）有限公司、南京数字金融产业研究院有限公司、苏州同济区块链研究院有限公司、中钞信用卡产业发展有限公司、杭州区块链技术研究院。

本文件主要起草人：姚前、王建平、罗凯、蒋东兴、李宇、蒋国庆、彭枫、王继伟、陈柏峰、周云晖、王凤冬、刘彬、王少清、杨博、马堃、李思颖、陈小泉、夏雄涛、王强、朱培、周耀亮、林智辰、葛浩、陶祖国、邵洪峰、孙北北、胡爽峰、肖东、奚海峰、曹恒、谷新萍、张业龙、赵滨、许明县、太贤美、陈莹、邵俊杰、迟云蔚、马小峰、万强、叶蔚、张一锋、王加楠。

引　言

区域性股权市场是我国资本市场的重要组成部分，是多层次资本市场体系的基石。区块链技术与区域性股权市场分散特征天然匹配，从新型金融基础设施层面为场外参与各方提供公共的可信服务，以技术手段完善市场基础条件，弥补区域性短板，解决登记效力不足、信用支撑不足、连通性和透明规范性不足等基础性问题，更好地发挥区域性股权市场的灵活优势，激发创新活力。基于监管链和地方业务链的双层架构，可以更好地支持区域性股权市场登记业务、交易报告库等业务和监管创新。监管链跨链对接各区域股权市场地方业务链，以监管链治理地方业务链，同时为地方业务链赋能，支持业务创新和监管创新。从建立逻辑统一、互联互通的区域性股权市场新型金融基础设施出发，有必要定义监管链与各地方业务链的跨链对接技术规范。

区域性股权市场区块链跨链技术规范

1 范围

本文件规定了区域性股权市场中地方业务链实现与监管链跨链对接的技术规范与系统实现要求，包括对应用环境、业务数据存储及地方业务链的节点接口要求等。

本文件适用于在区域性股权市场中进行地方业务链系统建设或服务运营的机构。

2 规范性引用文件

下列文件中的内容通过文中的规范性引用而构成本文件必不可少的条款。其中，注日期的引用文件，仅该日期对应的版本适用于本文件；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 25069 信息安全技术 术语

GB/T 32905-2016 信息安全技术SM3密码杂凑算法

GB/T 32918—2016 SM2椭圆曲线公钥密码算法

GM/T 0015—2012 基于SM2密码算法的数字证书格式规范

GM/T 0004-2012 SM3密码杂凑算法

JR/T 0184—2020 金融分布式账本技术安全规范

JR/T 0193—2020 区块链技术金融应用评估规则

3 术语和定义

GB/T 25069 界定的以及下列术语和定义适用于本文件。

3.1

区块链 blockchain

一种由多方共同维护，使用密码学保证传输和访问安全，能够实现数据一致存储、防篡改、防抵赖的技术体系。

[来源：JR/T 0193—2020，3.1]

3.2

区块 block

区块链中存储数据的单元。

[来源：JR/T 0193—2020，术语和定义3.2]

3.3

区块高度 block height

标识区块顺序的序号。

3.4

网（络） network

对各实体及其互连所做的一种安排。
[来源：GB/T 5271.18—2008，术语和定义2.18.01.01]

3.5

子网 subnetwork; subnet

在网络中，在元素间有一组共同特征，有明确界限，本身又能视为网络的那一部分。
[来源：GB/T 5271.18—2008，术语和定义2.18.01.05]

3.6

全节点 full node

负责账本数据完整性的节点，在有些区块链系统中也被称为记账节点。
[来源：JR/T 0184—2020，术语和定义3.25，有修改]

3.7

共识节点 consensus node

负责账本数据一致性的节点，在有些区块链系统中也被称为验证节点。
[来源：JR/T 0184—2020，术语和定义3.24，有修改]

3.8

状态 state

在给定的瞬间由定义系统、部件或模拟的特征的变量假定的值。
[来源：GB/T 11457—2006 信息技术 软件工程术语，术语和定义2.1568]

3.9

世界状态 world state

区块链上所有数据在某一时刻所表征的当前状态集合。

3.10

共识协议 consensus protocol

分布式账本系统中各节点为达成一致采用的计算方法。
[来源：JR/T 0184—2020，术语和定义3.17]

3.11

拜占庭容错 byzantine fault tolerance

在不可信环境中，即使系统部分组件失效或存在恶意角色，分布式系统仍然能够保持一致性要求的能力。

[来源：ITU-T X.1400-2020，6.11，有修改]

3.12

跨链 cross chain

实现在区块链之间的信息交互、信息验证与服务调用的技术。

3.13

交易 transaction

区块链上的一次原子性账本数据状态变更及其过程和结果记录，由交易标识符唯一标识。
[来源：ITU-T X.1400-2020，6.65，有修改]

3.14

共识状态 consensus state

共识协议在每个共识节点维护的特定区块的全局状态，是跨链协议所依赖的一个关键数据结构。
注：共识状态一般会包括最新的世界状态的哈希值、共识签名、共识节点公钥等元数据

3.15

默克尔树 merkle tree

一种其叶子节点标记数据的哈希值、每个非叶子节点标记其所有子节点的哈希值的树。
[来源：ISO 22739:2020，3.47]

3.16

默克尔证明 merkle proof

利用默克尔树来快速证明一段数据是否存在于某数据集合中的方法。

3.17

终局性 finality

交易一旦确认，就不会被回滚（rollback）或者撤销。
[来源：JR/T 0184—2020，术语和定义3.31]

3.18

状态根 state root

在区块中，包含该区块的世界状态信息的树根节点。

3.19

交易根 transaction root

在区块中，包含该区块的所有交易信息的树根节点。

3.20

监管链 global regulation blockchain

以全局服务和监管为目的构建的区块链系统。

3.21

地方业务链 regional business blockchain

各地方以服务区域性股权市场业务和生态为目的构建的区块链系统。

3.22

地方业务系统 regional business system

与地方业务链对接，实现数据传输到地方业务链的业务系统。
注：地方业务系统包括区域性股权交易中心业务系统等。

3.23

业务数据 business data

在地方业务链系统中运行的数据。

3.24

（业务）数据类型 business data type

业务数据按照主题和分类，组织成彼此相关联的类别。

3.25

数据模型 data model

对业务数据的标准化建模，一组标准的数据类型的定义。

3.26

数据对象 data object

地方业务链为每个数据类型生成的一个或多个以对象方式组织的实例。

3.27

数据对象引用 data object reference

为获取数据对象，存储在地方业务链上的标识信息，一般采用统一资源标识符的形式。

4 应用环境

区域性股权市场跨链技术规范的应用环境参考示例参见附录A。

5 业务数据存储

5.1 基本要求

业务数据在地方业务链存储，并可跨链至监管链。

每个业务数据在地方业务链生成一个或多个实例（即：业务数据对象）；而每个数据对象可以有一个或多个版本（即：业务数据对象快照或简称快照），代表对该数据对象执行创建/修改/删除/重写操作的结果。

5.2 业务数据对象编码

业务数据对象由数据头、数据体两部分组成，以JSON格式表示，采用UTF-8方式编码。

5.3 数据头

数据头应符合表1的要求，包含数据模型信息、数据对象信息、数据所有者签名、数据对象元数据等属性。签名者公钥、签名摘要算法采用的密码算法应符合符合GB/T32905-2016 和GB/T32918 -2016等相关国家标准以及GM/T0015—2012、GM/T0004-2012、JR/T 0184—2020等相关行业标准。

表 1 数据头格式

字段定义	字段路径	字段类型	字段长度
数据模型名称	header.model.protocol	字符串	64
数据模型版本	header.model.version	字符串	32
数据对象类型	header.content.type	字符串	32
数据对象标识	header.content.object_id	字符串	128
数据对象版本	header.content.version	整数型（初始版本号为 0，版本号依次递增）	
对应操作	header.content.operation	字符串（“create”：创建、“update”：修改、“overwrite”：覆盖、“delete”：删除）	16
时间戳（UNIX 时间）	header.content.timestamp	整数型	
备注	header.remark	字符串	2048
数据所有者签名	header.seal	JSON 对象集合	
签名者公钥类型	header.seal.pub_key.type	字符串（“SM2”）	64
签名者公钥值	header.seal.pub_key.value	字符串	512
签名摘要算法	header.seal.digest_algo	字符串（“SM3”）	32
签名	header.seal.signature	字符串	1024
数据对象元数据	header.metadata	JSON 对象	

具体可参考附录B数据对象数据头参考示例。

5.4 数据体

数据体应符合表2的要求，包含具体的数据内容。

表 2　数据体格式

字段定义	字段路径	字段类型
数据体内容	body	JSON 对象

5.5　数据对象生命周期管理

在关键业务流程执行的每个关键环节，所有被影响的业务数据需要同步体现在对应的链上数据对象中。数据对象的生命周期管理要满足：

a) 数据对象标识是数据对象全生命周期中的唯一标识，不随数据对象版本和内容的更新而变更；
b) 创建数据对象：
   1) 地方业务链应生成其系统唯一的对象标识；
   2) 以版本号 0 在链上存储完整快照，或链上存证并在链外存储完整快照；
c) 修改数据对象：
   1) 应在链上或链外存储数据体为全量或增量内容、数据头版本号加一的新快照；
   2) 若采用增量内容更新且要将上一版本中的某属性值修改为空值，无论该属性的类型是什么，都应将其值设置为 null；
   3) 增量更新需要满足：将增量内容合并至由所有历史版本组成的全量快照而得到的最新版本的全量快照，仍然可以通过 JSON Schema 的合法性校验；
d) 重写数据对象：
   1) 应在链上或链外存储数据体为全量或增量内容、数据头版本号与被重写快照一致的新快照；
   2) 若被重写的快照版本不是该对象的最新版本，则从该版本到最新版本之间所有的快照，都必须重写；
   3) 若采用增量内容重写且要将上一版本中的某属性值修改为空值，无论该属性的类型，都应将其值设置为 null；
   4) 增量重写需要满足：将增量内容合并至上一版本的全量快照而得到的最新版本的全量快照，仍然可以通过 JSON Schema 的合法性校验；
e) 删除数据对象：
   1) 应在链上或链外存储数据体为空、数据头版本号加一的新快照，历史快照不应删除；
   2) 删除操作是一个数据对象生命周期的终止操作，被执行过删除的数据对象，不应再有任何新的操作；
f) 无论创建、修改、重写还是删除操作，生成的对应快照都由“数据头”+“数据体”组成；
g) 数据对象引用：
   1) 若不同数据对象之间有引用关系，须按照业务发生时序，引用业务发生时对应的快照；“数据对象引用”属性的取值为：object_id/version；
   2) 当被引用的数据对象发生版本更新时，无需更新引用值；

具体可参考附录C数据对象生命周期管理参考示例。

业务数据须通过JSON Schema的合法性校验方能完成跨链处理，未能通过校验时将被作为异常记入跨链日志，地方业务链可通过监管链提供的跨链服务查询校验结果反馈作为更正参考。

5.6　数据对象存储

地方业务链可根据隐私保护的需求，将快照的部分内容记录在链上，同时将其全部内容通过链外存储服务在链外予以保存。链上记录应包含链外对应快照的哈希值。链外存储工具为兼容 AWS S3 协议的对象存储工具。

地方业务链须对行业监管机构监管链开放查询接口，该接口应满足以下要求：

——查询接口应通过 TLS 和鉴权加以保护。

——以数据对象标识和版本号为参数，在地方链上或链外可以获取到指定版本号的快照。

——查询返回的快照以统一的 JSON 格式编码（在快照上链之前，应使用数据对象 JSON Schema 进行合法性校验）。

具体参考可见附录D数据对象JSON Schema使用示例。

如果地方业务链暂时无法将所有业务数据实时记录在链上，作为临时过渡方案，可以将快照在链上存证，具体参考可见附录E数据对象快照参考示例。

6 地方业务链的节点接口要求

6.1 基本要求

地方业务链应按照监管链的要求提供节点接口，每个地方业务链有自定义的区块链网络标识。

地方业务链应采用拜占庭容错的等支持容错的共识协议。

地方业务链应向监管链暴露至少两个不同全节点的RPC接口。监管链将通过这些RPC接口获取地方业务链的链上数据。

地方业务链的节点RPC接口应支持监管链实现获取网络信息、区块信息、共识状态、业务数据等能力。

6.2 获取区块链网络信息的接口

获取区块链网络信息的接口用于获取可用网络或子网的列表、网络的当前状态（最新的终局性区块）以及监管链希望提供的其他有用信息。接口应符合表3的要求。

表 3 获取地方业务链网络信息的接口要求

接口定义	请求内容	响应内容
获取可用网络列表		可用网络列表，内容包括：区块链网络标识、区块链网络信息及其子网列表
获取网络配置选项	区块链网络标识	节点版本、允许的操作类型及其结果状态
获取当前网络状态	区块链网络标识	当前区块高度、区块哈希、区块时间戳、连接的节点等

6.3 获取区块信息的接口

获取区块信息的接口用于访问保存在区块里的所有数据。该接口应符合表4的要求，返回一个区块里所有改变链上世界状态的操作。

表 4　获取地方业务链区块信息的接口要求

接口定义	请求内容	响应内容
获取区块信息	区块链网络标识、区块高度	区块高度及其哈希、前一个区块的高度及其哈希、区块时间戳、包含的交易标识符、区块头等 每个区块的大小应不超过 16MB
获取交易信息	区块链网络标识、区块高度、交易标识符	交易包含的一系列操作，其操作类型及结果

6.4　获取共识状态信息的接口

该接口应符合表5的要求，以获取某个区块高度的共识状态。

表 5　获取地方业务链共识状态信息的接口要求

接口定义	请求内容	响应内容
获取某个区块高度的共识状态	区块链网络标识、区块高度	状态根、交易根以及其他信任信息，如：共识节点的公钥集、共识过程中的投票信息，包括，参与投票的共识节点列表、本区块的共识节点签名列表、签名数据（一般为区块哈希）等

6.5　获取业务数据对象或业务数据对象引用的接口

该接口应符合表6的要求，以获取业务数据对象或业务数据对象引用。如该接口返回的是业务数据对象引用，监管链应可通过该业务数据对象引用访问并获取到业务数据对象。

表 6　获取地方业务链业务数据对象的接口要求

接口定义	请求内容	响应内容
获取某一个区块包含的数据对象标识	区块链网络标识、区块高度	指定区块所包含的数据对象标识
获取数据对象或数据对象引用	区块链网络标识、区块高度、数据对象标识	数据对象在链上的实际编码内容及其默克尔证明 或数据对象引用（一般为数据对象的统一资源标识）

注：如地方业务链因底层技术原因无法提供默克尔证明，应提供可快速证明数据是否存在于某数据集合中的方法，且该方法须满足监管链的验证要求。

附 录 A
（资料性）
应用环境参考示例

图 A.1 是区域性股权市场跨链技术规范应用环境的参考示例。在该参考示例中，区域性股权市场跨链技术规范用于地方业务链与监管链之间数据对象的跨链处理。

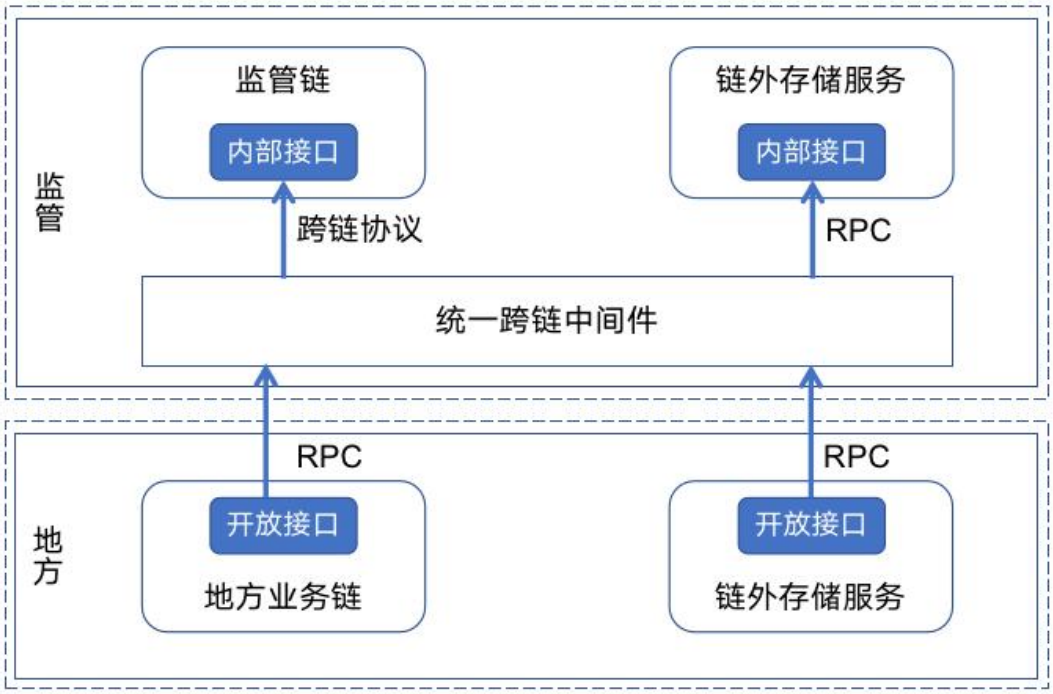


图 A.1 应用环境参考

附 录 B
（资料性）
数据对象数据头参考示例

以下是区域性股权市场跨链技术规范业务数据对象数据头的参考示例。在该参考示例中，区域性股权市场跨链技术规范规定了地方业务链的本地数据对象的数据头。


```
{
  "header": {
    "content": {
      "object_id": "goid09bef7bbbcd82a29c3512d3949185d4084312d427343bcb86c165590",
      "operation": "update",
      "timestamp": 1631016244953,
      "type": "product",
      "version": 1
    },
    "model": {
      "protocol": "区域性股权市场跨链业务数据模型",
      "version": "2.5.0"
    },
    "source": {

    }
  },
  "body": {

  }
}
```

附 录 C
（资料性）
数据对象生命周期管理参考示例

图C.1是区域性股权市场跨链技术规范数据对象生命周期管理的示例参考，在该参考示例中，区域性股权市场跨链技术规范规定了地方业务链的本地数据对象跨链至监管链全局数对象的全生命周期变更处理。

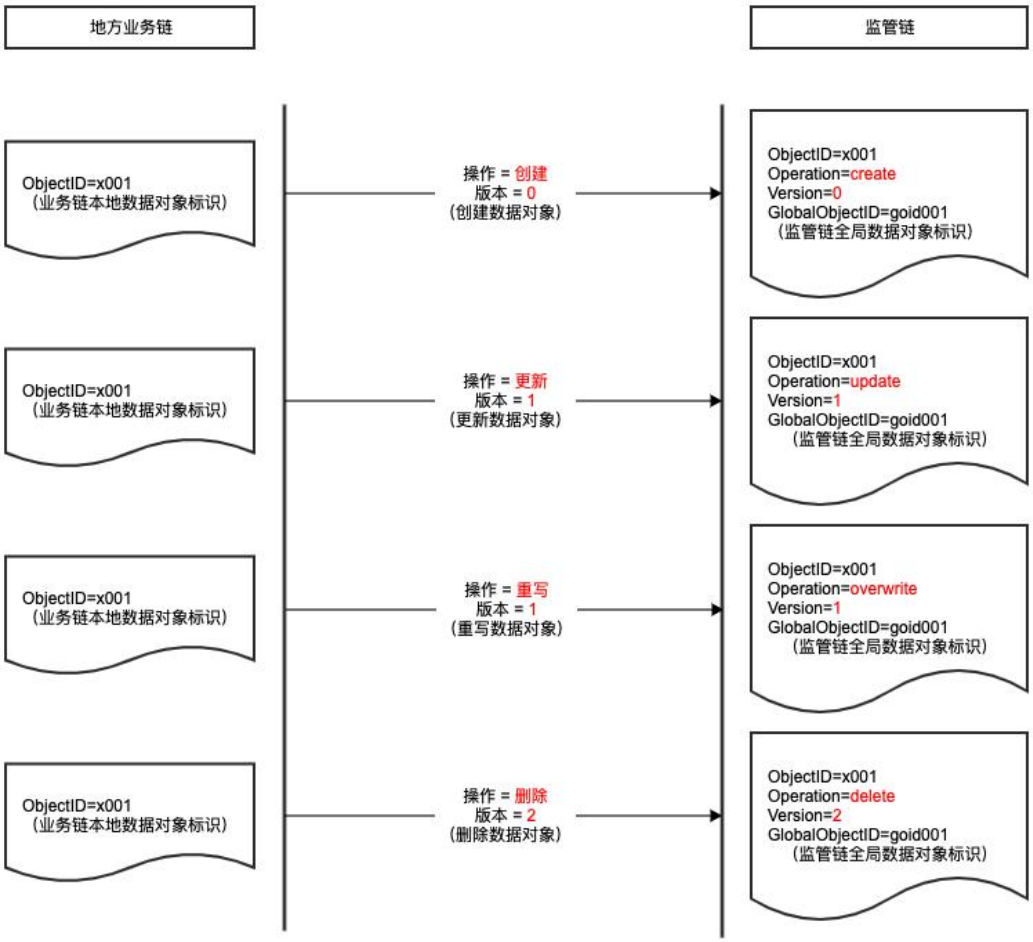


图 C.1 数据对象生命周期管理参考

附 录 D
（资料性）
数据对象 JSON Schema 参考示例

以下是区域性股权市场跨链技术规范对数据对象 JSON Schema 的参考示例。该参考示例仅用于演示 JSON Schema 的基本使用流程，不对 JSON Schema 工具的选择构成建议和要求，请根据自己的技术特点和需求进行评估。更多工具请参考：http://json-schema.org/implementations.html

在区域性股权市场区块链试点场景中，数据对象 JSON Schema 的基本使用流程为：

1) 使用 JSON Schema 生成代码
2) 基于 JSON Schema 生成的代码构造业务数据对象
3) 将业务数据对象序列化为 JSON 结构体（快照）
4) 使用 JSON Schema 校验序列化后的 JSON 结构体

使用工具生成业务数据对象代码，校验 JSON 结构及各属性值合法性的示例如下：

D.1 代码生成工具

工具网址：https://app.quicktype.io/#l=schema

通过以上网址在浏览器中打开在线代码生成工具，在网页左侧“Name”输入根对象名称“InterChainObject”，“Source type”选择“JSON Schema”，然后在文本域中输入完整的数据对象 Schema 内容；

在网页右侧悬浮框中，选择对应的开发语言等属性，即能够自动生成对应语言的代码。注意生成 Java 代码时：

——可能出现非必填日期/时间格式字段没有正确添加 JsonFormat 注解的情况，此系工具本身对 Java 语言代码生成支持存在缺陷导致，日期/时间格式字段需要手动在代码中添加；

——右侧选项框“Date time provider type”选择“Legacy”，则生成的代码中，日期会使用“java.utils.Date”类。

D.2 代码检验工具

D.2.1 在线格式校验

工具网址：https://www.jsonschemavalidator.net

使用示例如图D.1：

图 D.1　在线格式校验参考

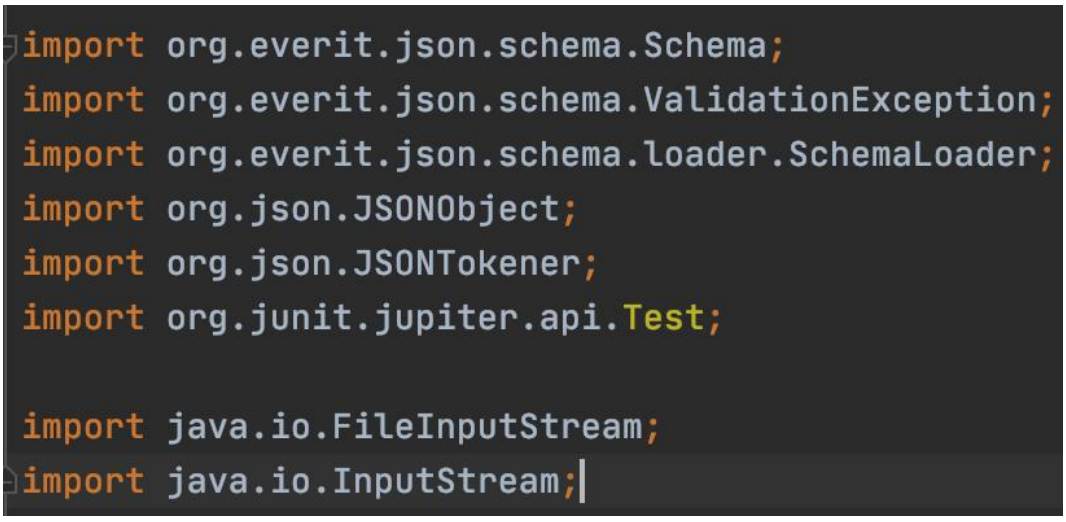
D.2.2　本地代码验证

GitHub: https://github.com/everit-org/json-schema

使用方法：

Maven 依赖可见上述 GitHub 链接

代码示例如图 D.2：



图D.2　本地格式校验参考

```java
public class ToolTest {
    @Test
    void testTool() {
        try (
                InputStream schemaInputStream = new FileInputStream( name: "demo.json");
                InputStream jsonInputStream = new FileInputStream( name: "product.json")
        ) {
            JSONObject rawSchema = new JSONObject(new JSONTokener(schemaInputStream));
            JSONObject rawJson = new JSONObject(new JSONTokener(jsonInputStream));
            Schema schema = SchemaLoader.load(rawSchema);
            schema.validate(rawJson); // throws a ValidationException if this object is invalid
        } catch (ValidationException e) {
            System.out.println(e.getErrorMessage());
            for(String s : e.getAllMessages()){
                System.out.println(s);
            }
            return;
        } catch (Exception e) {
            e.printStackTrace();
            return;
        }
        System.out.println("valid");
    }
}
```

图 D.2　本地格式校验参考（续）

当 JSON 格式或其属性值不符合 JSON Schema 中定义的规范时，会得到如图 D.3 的报错信息，通过报错信息可以轻松定位到错误原因并进行修复：

```
10 schema violations found
#/header/model/version: 2.0 is not a valid enum value
#/header/content: required key [version] not found
#/header/content/operation: add is not a valid enum value
#/body: required key [account] not found
#/body: required key [product] not found
#/body: required key [transaction] not found
#/body: required key [settlement] not found
#/body: required key [registration] not found
#/body/subject/basic_information_subject/general_information_subject: required key [subject_type] not found
#/body/subject/basic_information_subject/subject_qualification_information/0/investor_suitability_information/0/subject_investor_qua
```

图 D.3　校验报错信息参考

附 录 E
（资料性）
数据对象快照存证参考示例

以下是区域性股权市场跨链技术规范对数据对象快照存证的参考示例。在该参考示例中，地方业务链暂时无法将所有业务数据实时记录在链上，按以下步骤采用链上存证的临时过渡方案：

a) 在关键业务流程执行完毕时，按照业务逻辑触达的先后顺序生成所有被影响的快照，并将它们存储于链外（如：OSS）；
b) 每个链外存储成功的快照都要在链上存证；
c) 存证记录应该至少包含如下属性（以 JSON 格式为例）：

```
{
    "creator":"0xce29041d5f50993040b2cb3a5bd9fe0ac35b8de8",
    "data":[
        {
            "digest": "745B5F89......",    // 快照的摘要
            "digest_algo": "sha-256",      // 摘要算法
            "uri": "C0332F2C......",       // 快照的URI
            "meta": ""
        },
        {...}
    ]
}
```

每个存证记录可包含多个数据项，每个数据项对应一个快照的链上存证。