00A0 $\exists$ $\forall$ $\subseteq$ 2713x 27FA$\Longleftrightarrow$ $\surd$ $\sqrt[3]{}$ $\oplus$ $\otimes$

Weldentity Documentation

Junqi Zhang

2020 08 04

Contents:

CHAPTER 1

WeIdentity

1.1 WeIdentity

WeIdentity WeIdentity

1.1.1 1.

WeIdentity WeIdentity DID WeIdentity Credential

(WeIdentity DID)

WeIdentity DID FISCO-BCOS W3C DID WeIdentity DID Entity ID

WeIdentity DID

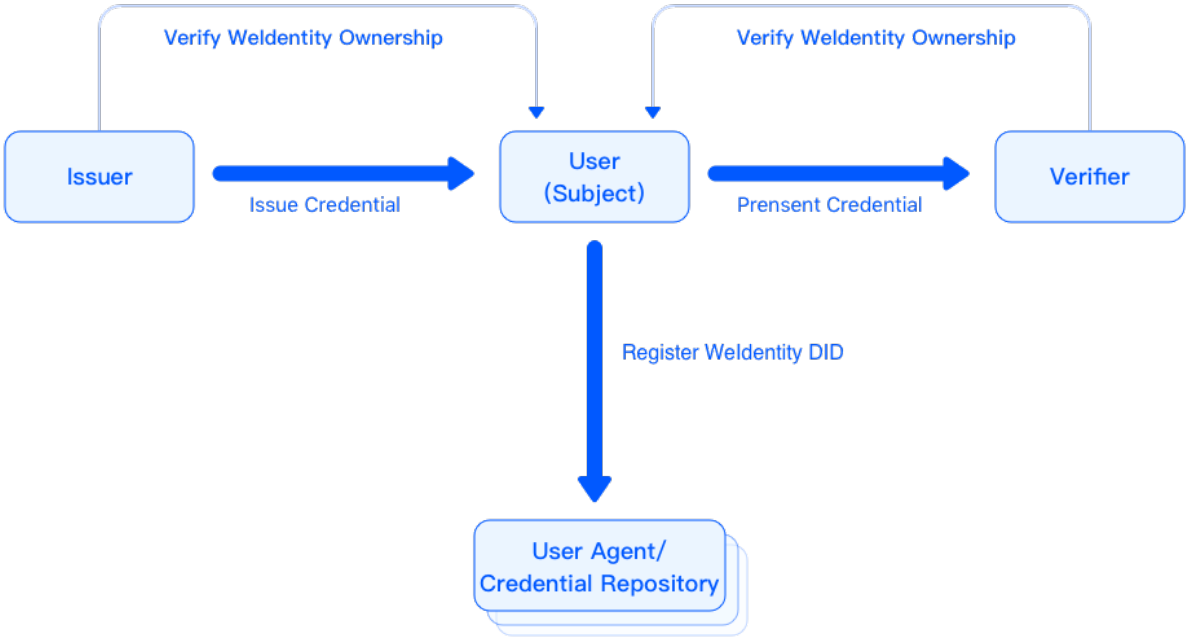
(WeIdentity Credential)

WeIdentity Credential W3C VC Credential

WeIdentity

WeIdentity WeIdentity Demo WeIdentity WeIdentity SDK

1.1.2 2. WeIdentity



WeIdentity

User (Entity)	WeIdentity DID Issuer Credential Verifier
Issuer	Credential WeIdentity DID Credential
Verifier	Credential WeIdentity DID Credential
User Agent / Credential Repository	WeIdentity DID Credential

WeIdentity User Agent DID Issuer DID Credential Credenti
+ Verifier

1.1.3 3. Demo

WeIdentity Demo

		WeID ID Hash

1.1.4 4.

WeIdentity FISCO-BCOS Java SDK RestService

1.1.5 5. Getting Started

WeIdentity

1.1.6

weidentity@webank.com

1.1.7

-

WeIdentity

WeIdentity WeIdentity

1.

WeIdentity WeIdentity DID WeIdentity Credential

(WeIdentity DID)

WeIdentity DID FISCO-BCOS W3C DID WeIdentity DID Entity ID

WeIdentity DID

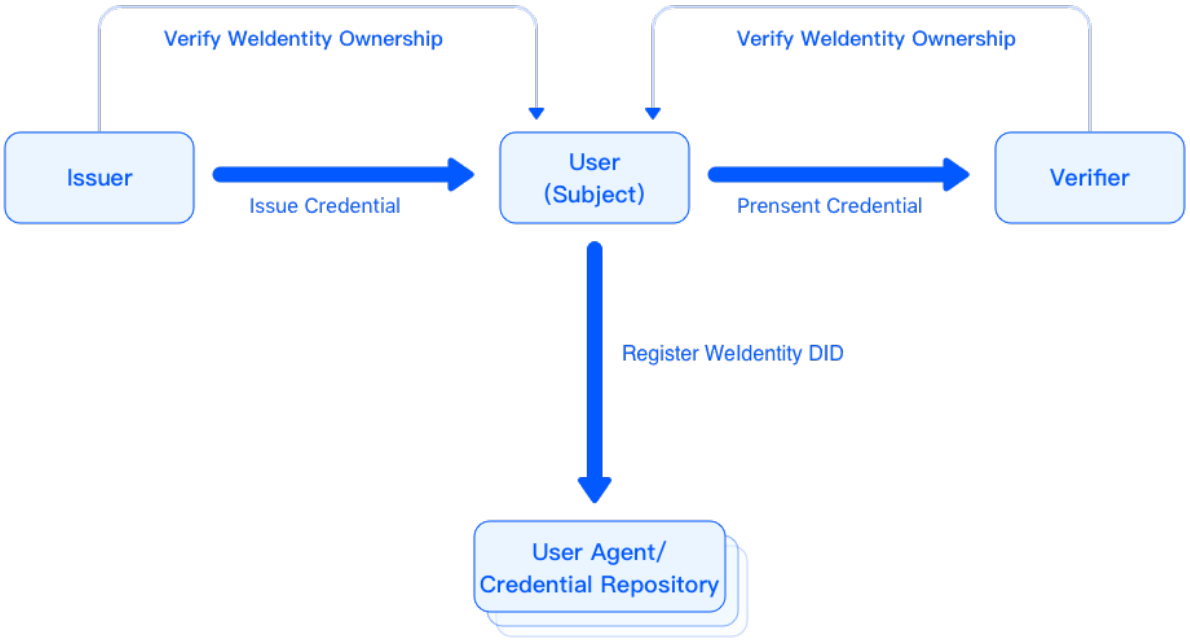
(WeIdentity Credential)

WeIdentity Credential W3C VC Credential

WeIdentity

WeIdentity WeIdentity Demo WeIdentity WeIdentity SDK

2. WeIdentity



WeIdentity

User (Entity)	WeIdentity DID Issuer Credential Verifier
Issuer	Credential WeIdentity DID Credential
Verifier	Credential WeIdentity DID Credential
User Agent / Credential Repository	WeIdentity DID Credential

WeIdentity User Agent DID Issuer DID Credential Credenti
+ Verifier

3. Demo

WeIdentity Demo

		WeID ID Hash

4.

WeIdentity FISCO-BCOS Java SDK RestService

5. Getting Started

WeIdentity

weidentity@webank.com

-

WeIdentity

1 WeIdentity

WeIdentity WeIdentity

2 WeIdentity

3 Sample

:	Sample

WeIdentity

4 Java Service WeIdentity Java SDK

WeIdentity Java SDK

:	Java RestService WeIdentity Rest Service

WeIdentity Java SDK

Java SDK WeIdentity

WeIdentity JAVA SDK

weid-java-sdk WeIdentity JAVA SDK

JSON-LD		JSON-LD JSON-LD Wiki
Authorization		Authorization vs Authentication
Authentication		Authorization vs Authentication
Public Key		Public Key Wiki

1 –

Private Key		Private Key Public Key Wiki
Signature		Digital Signature Wiki
DID		W3C DID ID W3C DID
WeIdentity DID		WeID WeIdentity ID ID W3C DID
WeIdentity Document		DID 3 Authentication Authenticate Service Endpoint
Claim		Credential Claim
WeIdentity Credential		“ ” W3C Verifiable Credential Credential Claim
Verifiable Credential		
Credential		
Notification		WeIdentity
CPT		Claim Protocol Type, Issuer Claim Claim CPT
chain-id	ID	WeIdentity owner WeIdentity
Service Endpoint		Entity Credential
publish		Issuer Authority Issuer CPT publish
issue		Issuer Authority Issuer CPT Credential issue
payload		Notification
Trusted Data		
Issuer		WeIdentity DID Entity Issuer Credential
Authority Issuer		Claim WeID
Entity		WeIdentity Document Credential WeIdentity DID
Verifier		Verifier
Data Repository		Credential APP
User Agent		APP
Publisher	CPT	CPT Publisher
Committee Member		Authority Issuer
Specific Issuer		Authority Issuer
Verifiable Presentation		W3C Verifiable Presentation Presentation Credential
Policy		Verifiable Presentation Credential Credential Claim

WeIdentity

WeIdentity DID Credential

WeIdentity

1.
2. KYC WeIdentity DID
3. Credential

-
-
  –
  –
  –
  –

- WeIdentity

1. WeIdentity DID KYC
2. Credential Credential
3. Credential Credential
4. Credential
5. Verify Credential
6. offer

-

-
  - –
  - –
  - –
  - –

- WeIdentity

1. WeIdentity DID KYC
2. Credential Credential
3.
4. Credential
5. Verify
6. Credential Verify
7.
8. Verify

Claim

name		gender	
department		drugs	
syndrome		diagnosis	
doctor	DID	hospital	DID

-

18

-
  - –
  - –

- WeIdentity

1. WeIdentity DID KYC
2. KYC Credential DID
3. Verify
4.

- A B WeIdentity
-
  - –
  - –
  - –
  - –
- WeIdentity

1. WeIdentity DID KYC
2. Credential WeIdentity DID
3. Credential
4. Verify
5.

- WeIdentity
-
  - –
  - –
  - –
  - –
- WeIdentity

1. WeIdentity DID
2. KYC WeIdentity DID
3. WeIdentity DID
4. KYC
5.
6.

- WeIdentity
- – – – –
- WeIdentity

1. WeIdentity DID
2. KYC WeIdentity DID
3.
4. Credential
5.
6. Verify
7.

WeIdentity

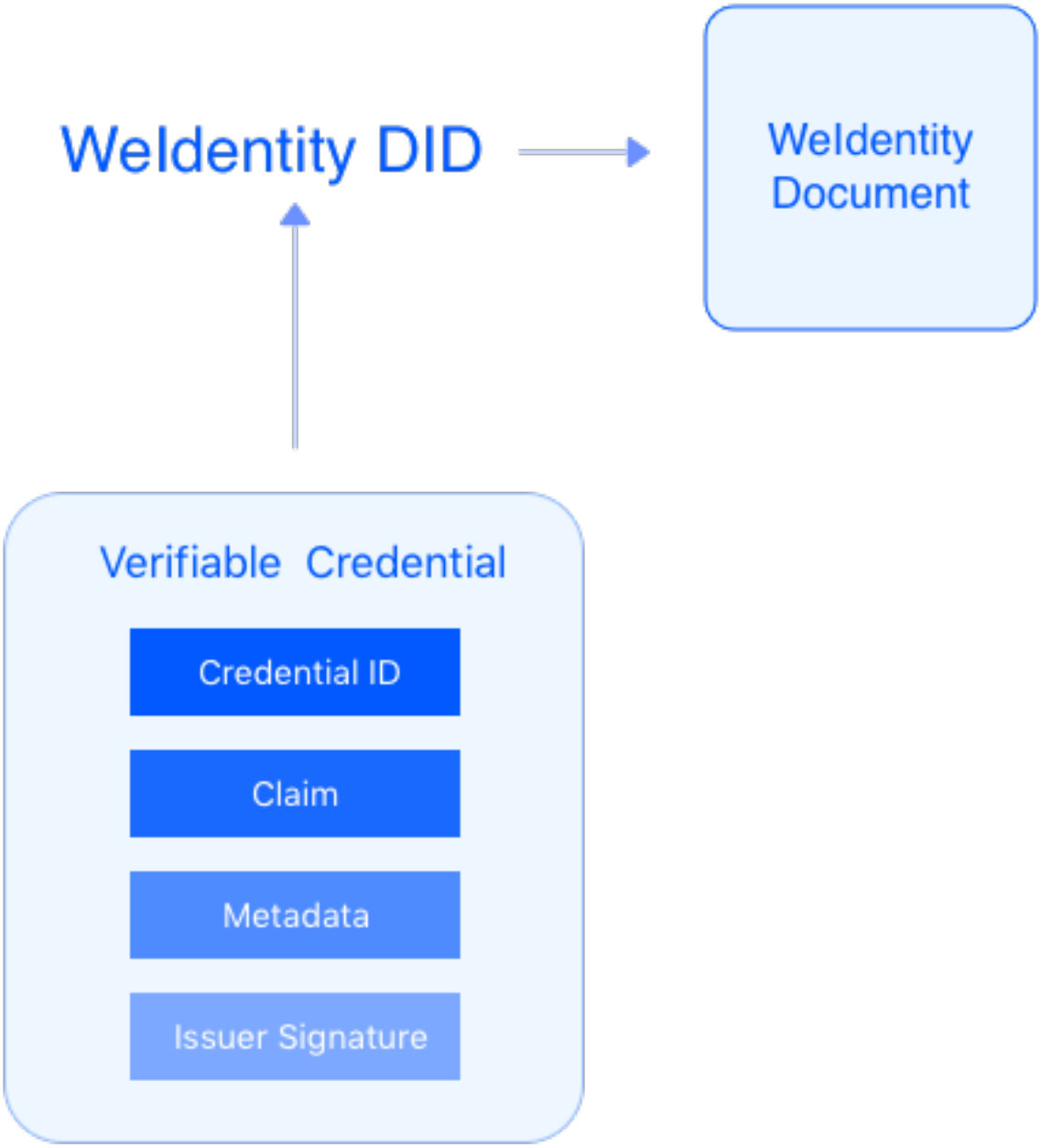
V0.1.0		
V0.2.0	ER	
V0.3.0	Notification	
V0.3.1	WeIdentity Document CPT	
V0.3.2		
V0.3.3		

1.

2.

3. WeIdentity DID

WeIdentity DID WeIdentity Credential



WeIdentity DID WeIdentity Credential WeIdentity DID WeIdentity Credential WeIdentity DID WeIdentity Credential WeIdentity Credential WeIdentity DID WeIdentity DID WeIdentity Document DID WeIdentity Credential

WeIdentity

1. KYC
2. WeIdentity DID
3. Credential
4. Issuer Credential
5. Credential Holder
6. Credential
7. Verifier Credential

WeIdentity DID

WeIdentity DID = did:weid:chain-id:bs-specific-string

did	DID “did”
weid	WeIdentity DID method name “weid”
chain-id	ID WeIdentity owner WeIdentity
bs-specific-string	Entity

bsSpecificString

(chain-id 101): `"did:weid:101:0x0086eb1f712ebc6f1c276e12ec21"`

WeIdentity Document

@context	WeIdentity Document
id	WeIdentity DID Document Entity
created	Document
updated	Document
publicKey	
publicKey.id	ID
publicKey.type	signature suite
publicKey.owner	Entity WeIdentity owner Document id Credential Entity owner
authentication	Entity Document
authentication.type	signature suite
authentication.publicKey	publicKey
service	service DID
service.id	service endpoint ID
service.type	service endpoint
service.serviceEndpoint	serviceEndpoint URI JSON-LD
service.	
recovery	WeIdentity DID WeIdentity

- WeIdentity DID Authorization Recovery

```
{
  "@context": "https://weidentity.webank.com/did/v1",
  "id": "did:weid:1:123456789abcdefghi",
  "created": "2017-09-24T17:00:00Z",
  "updated": "2018-09-24T02:41:00Z",
  "publicKey": [{
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "RsaVerificationKey2018",
    "owner": "did:example:123456789abcdefghi",
    "publicKeyPem": "-----BEGIN PUBLIC KEY...END PUBLIC KEY-----\r\n"
  }, {
    "id": "did:example:123456789abcdefghi#keys-2",
    "type": "Secp256k1VerificationKey2018",
    "owner": "did:example:123456789abcdefghi",
    "publicKeyHex": "02b97c30de767f084ce3080168ee293053ba33b235d7116a3263d29f1450936b71"
  }],
  "authentication": [{
    // this key can be used to authenticate as DID ...9938
    "type": "RsaSignatureAuthentication2018",
    "publicKey": "did:weid:1:123456789abcdefghi#keys-1"
    }
  ],
  "service": [{
    "type": "drivingCardService",
    "serviceEndpoint": "https://weidentity.webank.com/endpoint/8377464"
    }, {
    "type": "padiCertificateService",
    "serviceEndpoint": "https://weidentity.webank.com/endpoint/8377465"
  }],
  "recovery": ["did:weid:1:2323e3e3dweweewew2","did:weid:1:2323e3e3dweweewew3"],
}
```

WeIdentity DID

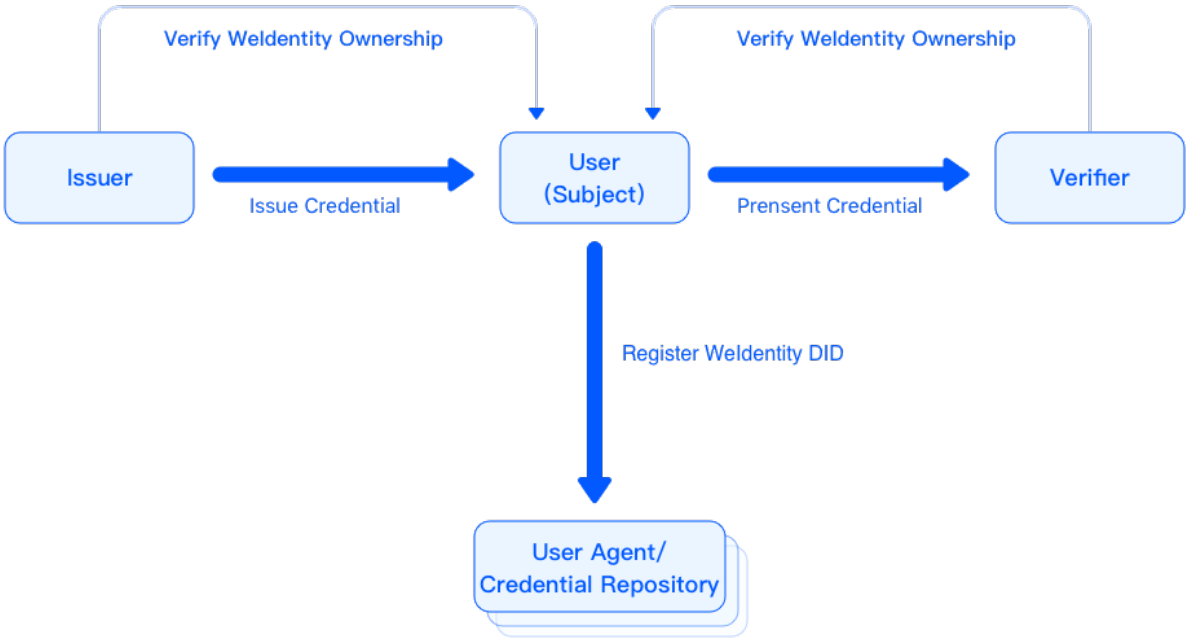
WeIdentity DID WeIdentity Document

/

WeIdentity DID WeIdentity Document

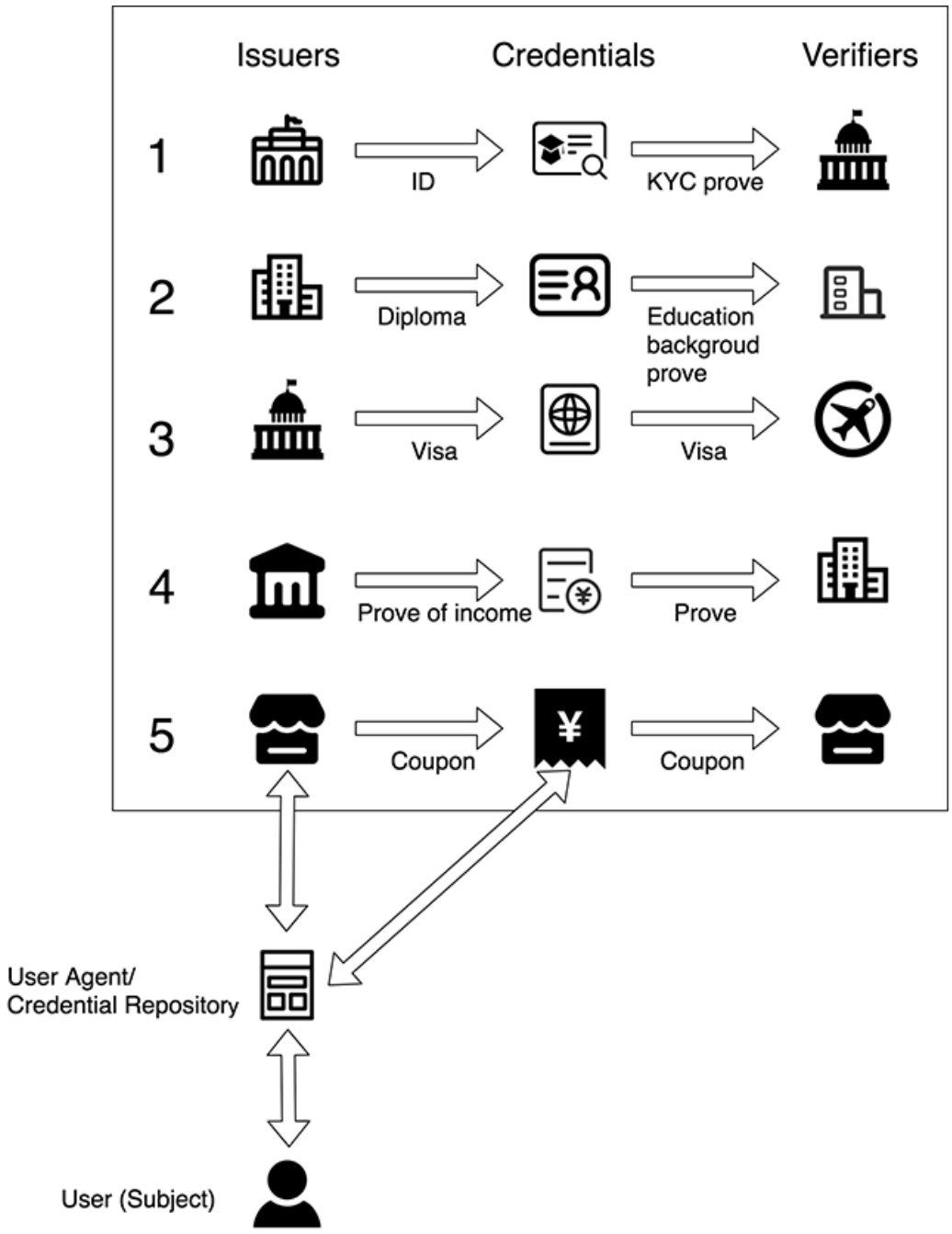
WeIdentity Document

4. WeIdentity Credential



WeIdentity

User (Entity)	WeIdentity DID Credential
Issuer	Credential WeIdentity DID Credential
Verifier	Credential WeIdentity DID Credential
User Agent / Credential Repository	WeIdentity DID Credential



WeIdentity Credential

1. Issuer Credential Verifier
2. Issuer Credential Verifier
3. Issuer Credential Verifier
4. Issuer Credential Verifier
5. Issuer Credential Verifier

Credential

@context	Credential
id	Credential ID UUID
issuer	Issuer DID WeIdentity
issued	issue
claim	Claim CPT CPT
claim.primeNumberIdx	index
claim.type	Claim Protocol Type ID CPT100
revocation	
signature	Issuer holder issuer
signature.type	
signature.created	
signature.creator	WeIdentity DID
signature.domain	domain
signature.nonce	
signature.signatureValue	value Credential signature

Credential

id	
type	
issued	
claimHash	Claim hash
revocation	
signature	

```
{
  "@context": "https://weidentity.webank.com/vc/v1",
  "id": "dsfewr23sdcsdfeqeddadfd",
  "type": ["Credential", "cpt100"],
  "issuer": "did:weid:1:2323e3e3dweweewew2",
  "issued": "2010-01-01T21:19:10Z",
  "claim": {
    "primeNumberIdx":"1234"
    //the other properties in this structure varied according to different CPT
  },
  "revocation": {
    "id": "did:weid:1:2323e3e3dweweewew2",
    "type": "SimpleRevocationList2017"
  },
  "signature": [{
    "type": "LinkedDataSignature2015",
    "created": "2016-06-18T21:19:10Z",
    "creator": "did:weid:1:2323e3e3dweweewew2",
    "domain": "www.diriving_card.com",
    "nonce": "598c63d6",
    "signatureValue": "BavEll0/I1zpYw8XNi1bgVg/sCneO4Jugez8RwDg/+MCRVpjOboDoe4SxxKjkC
  OvKiCHGDvc4krqi6Z1nOUfqzxGfmatCuFibcC1wpsPRdW+gGsutPTLzvueMWmFhwYmfIFpbBu95t501+r
    SLHIEuujM/+PXr9Cky6Ed+W3JT24="
  }]
}
```

Claim Protocol Type CPT

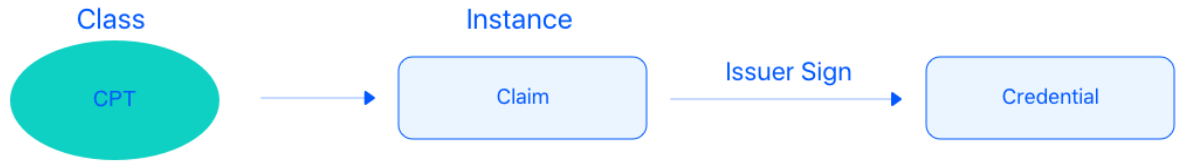
Issuer Claim Claim CPT CPT JSON-LD CPT

- CPT

@context	CPT
id	CPT
version	CPT
publisher	CPT WeIdentity DID
signature	CPT
claim	Claim
created	
updated	
description	CPT

- CPT

```
"CPT": {
  "@context" : "https://weidentity.webank.com/cpt100/v1",
  "version" : "v1",
  "id" : "CPT100",
  "publisher" : "did:weid:1:2323e3e3dweweewew2",
  "signature" : "BavEll0/I1zpYw8XNi1bgVg/sCneO4Jugez8RwDg/+MCRVpjOboDoe4SxxKjkC
  OvKiCHGDvc4krqi6Z1nOUfqzxGfmatCuFibcC1wpsPRdW+gGsutPTLzvueMWmFhwYmfIFpbBu95t501+r
    SLHIEuujM/+PXr9Cky6Ed+W3JT24=",
  "claim" : "",
  "address" : " ",
  "class" : "C1",
  "created" : "2010-06-20T21:19:10Z",
  "updated" : "2016-06-20T21:19:10Z",
  "description" : "    "
}
```



CPT Claim Claim CPT Issuer Claim Credential

Claim

Claim

Credential

Credential

issue Credential

Credential

Entity Credential

/ Credential

Credential holder Credential Credential

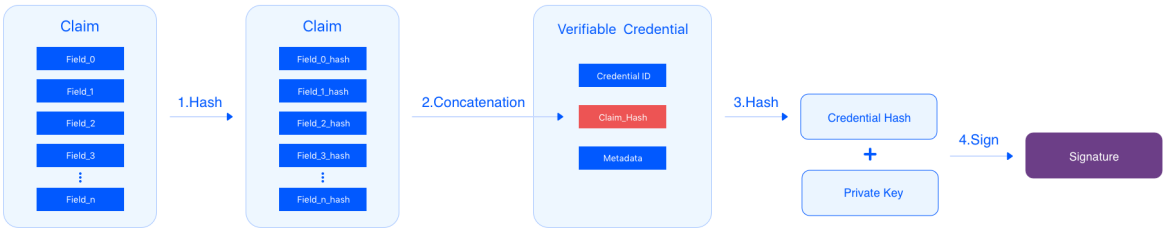
Credential

Credential Issuer Credential

Credential

Issuer Credential

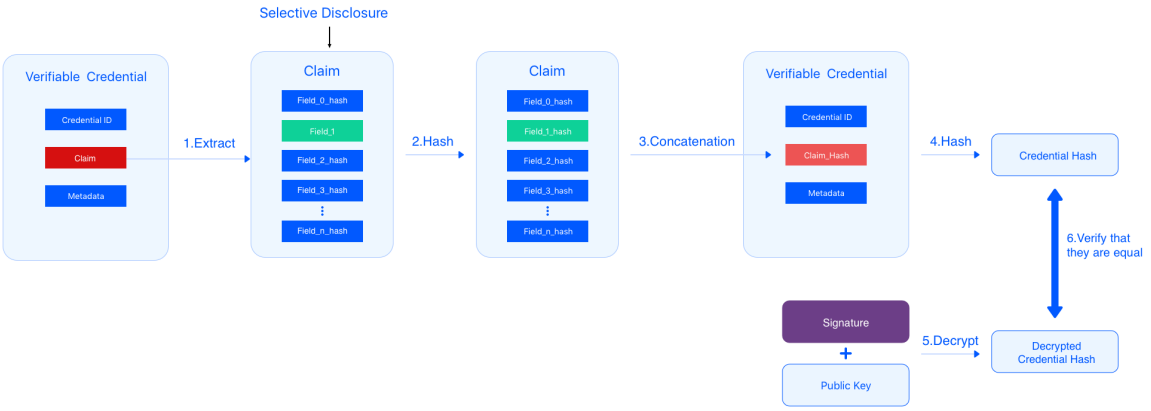
1. Claim hash
2. Claim hash Claim_Hash Credential hash Credential
3. Claim_Hash Credential hash Credential Hash
4. Private Key Credential Hash Signature



Credential

Field_1 hash Claim Credential Verifier Verifier

1. Verifier Credential Claim
2. Verifier hash Field_1, Field_1_hash , hash Claim
3. Claim hash Claim_Hash Credential hash Credential
4. Claim_Hash Credential hash Credential Hash
5. Credential Signature Issuer public key decrypt
6. Credential Signature Credential



Credential

Credential

1. 1 a a a
2. semiprime

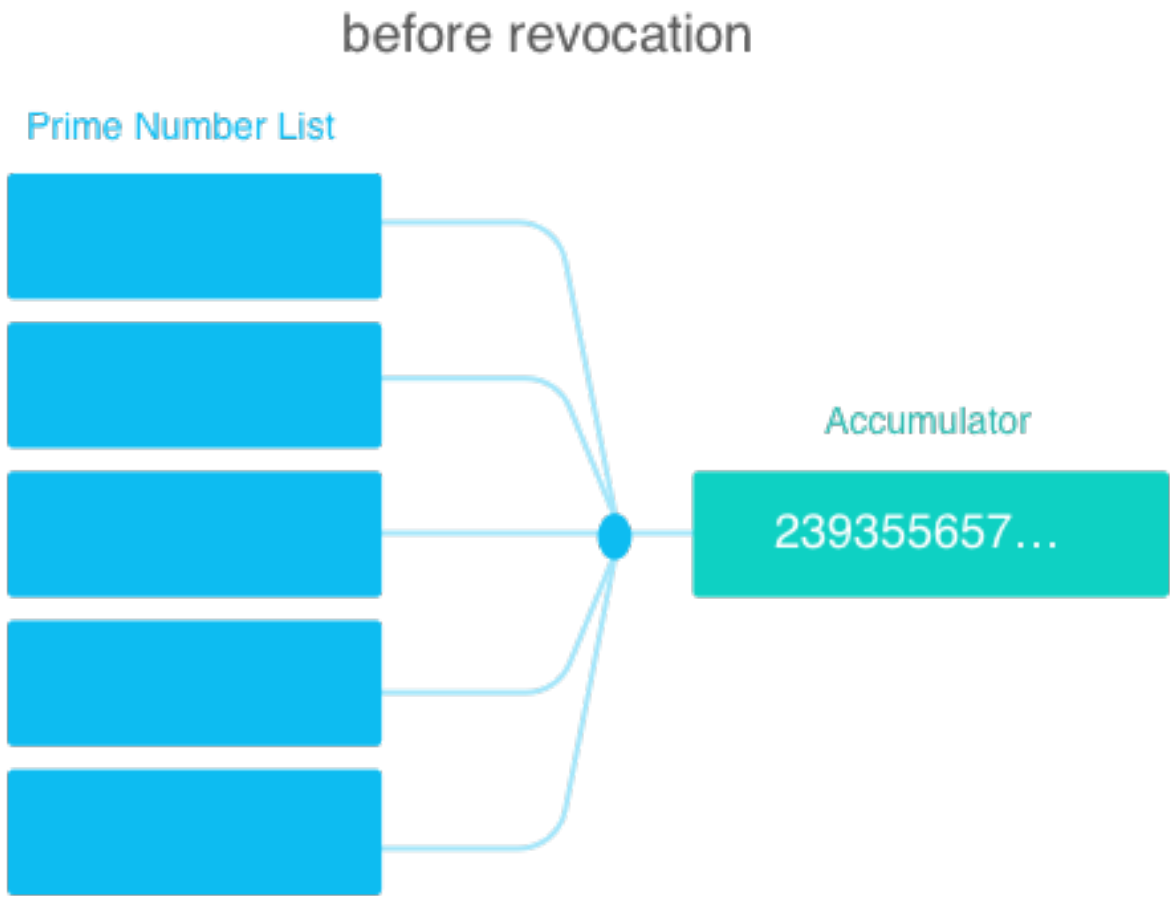
WeIdentity index Issuer Issuer Credential index Credential Credential Accumulator Issuer Accumulator Accumulator

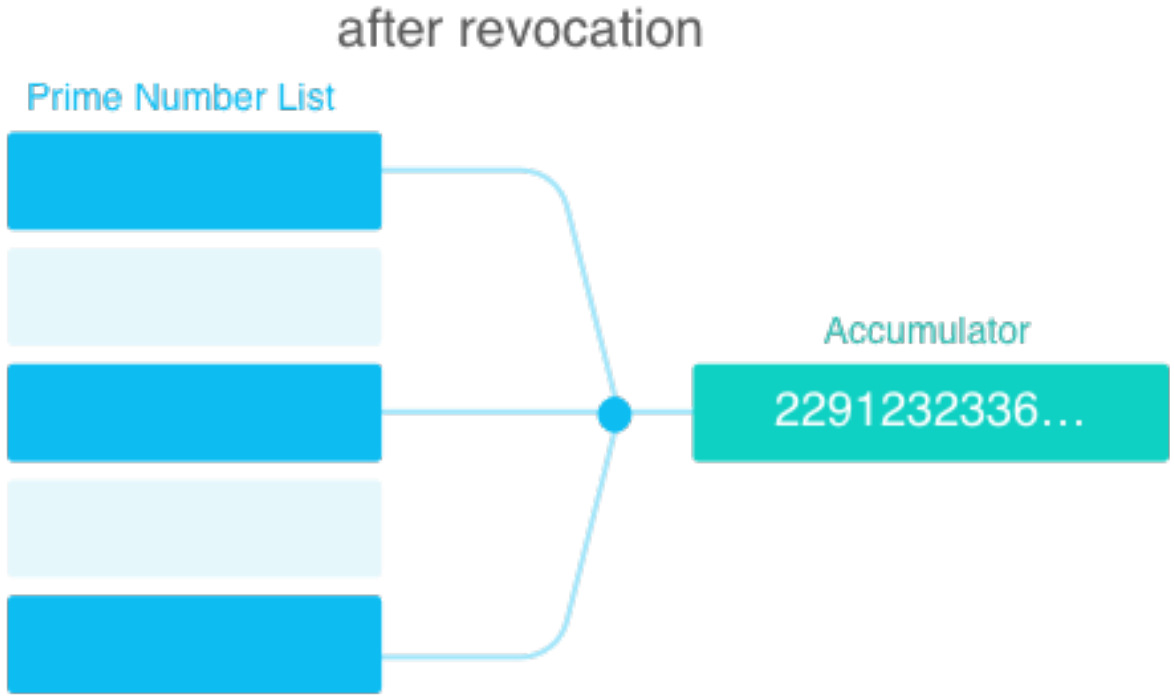
Issuer Credential

Credential Issuer Accumulator Accumulator

Verifier Credential

Credential Credential Issuer Accumulator Credential





5. Authority Issuer

WeIdentity DID issue Claim Authority Issuer Authority Issuer

version	version
id	Authority Issuer WeIdentity DID WeIdentity ID
name	Authority Issuer
created	
updated	
publicKey	Authority Issuer
validCrenRef	Authority Issuer credential primeNumber

6. Notification

type notification payload

type	register, update
weid	notification WeIdentity DID
payload	Notification

WeIdentity DID

WeIdentity DID notification WeIdentity DID

type	register

- payload

NULL

WeIdentity DID

WeIdentity DID notification public key meta data

type	document_mod WeIdentity Document

- payload payload

operation	add, update remove
field	key
original	
new	

Notification

Credential payload

type	transportation

Notification

7.

- W3C DID Spec
- W3C Verifiable Credentials
- Linked Data Signatures 1.0 Draft
- RSA Signature Suite 2018

WeIdentity Sample

weid-sample WeIdentity Java WeIdentity Java

WeIdentity JAVA SDK

weid-sample

-
- spring-boot

WeIdentity WeIdentity

- Issuer

  WeID
  Authority Issuer
  CPT
  Credential

- User Agent

  WeID
  Presentation
  Presentation QRcode Json Verifier

- Verifier

  User Agent Presentation
  Presentation

WeIdentity-Sample WeIdentity WeIdentity JAVA SDK Java weid-sample

1.

1.1 WeIdentity-Sample

```
git clone https://github.com/WeBankFinTech/WeIdentity-Sample
```

```
: git clone https://gitee.com/WeBank/WeIdentity-Sample
```

1.2

WeIdentity Sample WeIdentity Build Tool weid-sample Build Tool

WeIdentity Sample, weid-java-sdk

- WeIdentity-Sample

  WeIdentity-Sample

```
chmod +x *.sh
./build.sh
```

2.

- Issuer

```
./command.sh issuer
```

WeID Authority Issuer CPT Credential

```
--------- start issuer ----------
issuer() init...

begin to createWeId...

createWeId result:

result:(com.webank.weid.protocol.response.CreateWeIdDataResult)
weId: did:weid:1:0x7a276b294ecf0eb7b917765f308f024af2c99a38
userWeIdPublicKey:(com.webank.weid.protocol.base.WeIdPublicKey)
    publicKey:␣
↪1443108387689714733821851716463554592846955595194902087319775398382966796515741745
    95118210554711531306779199915498227256788151940687396693589185508570578 4
userWeIdPrivateKey:(com.webank.weid.protocol.base.WeIdPrivateKey)
    privateKey: 46686865859949148045125507514815998920467147178097685958028816903332430030079
errorCode: 0
errorMessage: success
transactionInfo:(com.webank.weid.protocol.response.TransactionInfo)
blockNumber: 2098
transactionHash: 0x20fc5c2730e4636248b121d31ffdbf7fa12e95185068fc1dea060d1afa9d554e
transactionIndex: 0

begin to setPublicKey...

setPublicKey result:

result: true
errorCode: 0
errorMessage: success
transactionInfo:(com.webank.weid.protocol.response.TransactionInfo)
blockNumber: 2099
transactionHash: 0x498d2bfd2d8ffa297af699c788e80de1bd51c255a7365307624637ae5a42f3a1
transactionIndex: 0
```

- User Agent

```
./command.sh user_agent
```

WeID Presentation Presentation QRcode Json

```
--------- start User Agent ----------
userAgent() init...

begin to create weId for useragent...

createWeId result:

result:(com.webank.weid.protocol.response.CreateWeIdDataResult)
weId: did:weid:1:0x38198689923961e8ecd6d57d88d027b1a6d1daf2
userWeIdPublicKey:(com.webank.weid.protocol.base.WeIdPublicKey)
    publicKey:␣
↪1240951307719395926589625269367299070161485161875394060374281929079442269004878 6166
```

(continues on next page)

( )

```
    77748624449230242365328258533877448834753636236821653645295685212386945 6
userWeIdPrivateKey:(com.webank.weid.protocol.base.WeIdPrivateKey)
    privateKey: 11700070604387246310492373601720779844791990854359896181912833510050901695117
errorCode: 0
errorMessage: success
transactionInfo:(com.webank.weid.protocol.response.TransactionInfo)
blockNumber: 2107
transactionHash: 0x2474141b82c367d8d5770a7f4d124aeaf985e7fa3e3e2f7f98eeed3d38d862f5
transactionIndex: 0
```

- Verifier

```
./command.sh verifier
```

Verifier Presentation Presentation

```
--------- start verifier ----------
verifier() init...

begin get the presentation json...
```

WeIdentity-Sample WeIdentity-Sample `com.webank.weid.demo.command.DemoCommand` Java

spring-boot

`spring-boot` `weid-sample` `swagger`

1.

```
git clone https://github.com/WeBankFinTech/weid-sample.git
```

2.

2.1 weid-java-sdk

WeIdentity Sample springboot WeIdentity Build Tool

WeIdentity Sample, weid-java-sdk

2.2

2.2.1

- weid-sample

```
cd weid-sample
chmod +x *.sh
./build.sh
```

- weid-sample :

```
./start.sh
```

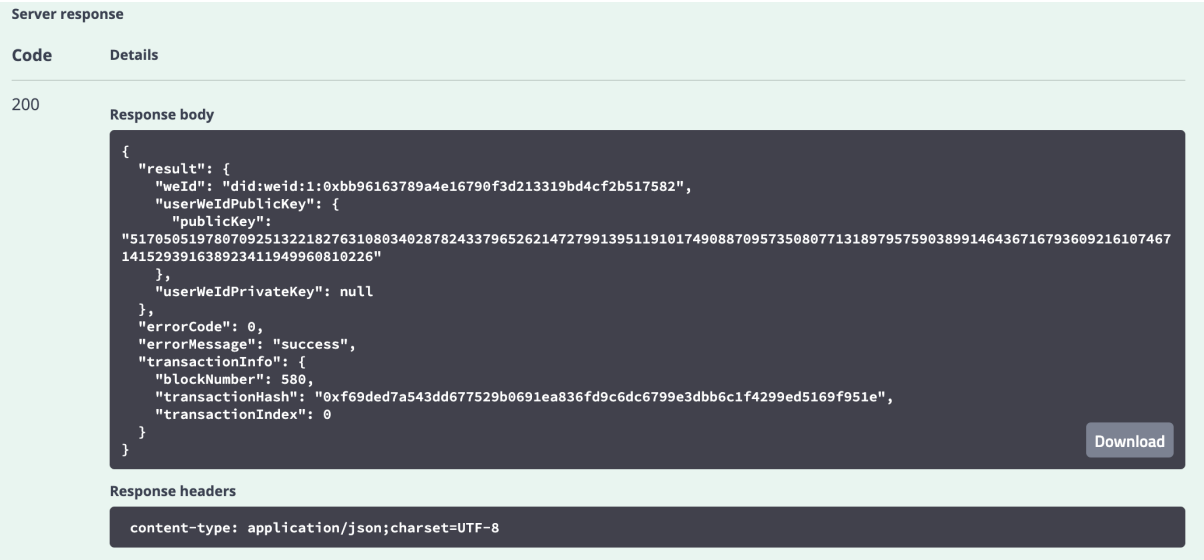
```
[main] INFO  AnnotationMBeanExporter() - Registering beans for JMX exposure on startup
[main] INFO  Http11NioProtocol() - Initializing ProtocolHandler ["https-jsse-nio-6101"]
[main] INFO  Http11NioProtocol() - Starting ProtocolHandler ["https-jsse-nio-6100"]
[main] INFO  NioSelectorPool() - Using a shared selector for servlet write/read
[main] INFO  Http11NioProtocol() - Initializing ProtocolHandler ["http-nio-6101"]
[main] INFO  NioSelectorPool() - Using a shared selector for servlet write/read
[main] INFO  Http11NioProtocol() - Starting ProtocolHandler ["http-nio-6101"]
[main] INFO  TomcatEmbeddedServletContainer() - Tomcat started on port(s): 6100 (https) 6101␣
↪(http)
[main] INFO  SampleApp() - Started SampleApp in 3.588 seconds (JVM running for 4.294)
```

2.2.2

127.0.0.1 6101 resources/ http://127.0.0.1:6101/swagger-ui.html WeIdentity

- WeID

‘‘/step1/issuer/createWeId‘‘ WeID



WeID did:weid:1:0xbb96163789a4e16790f3d213319bd4cf2b517582

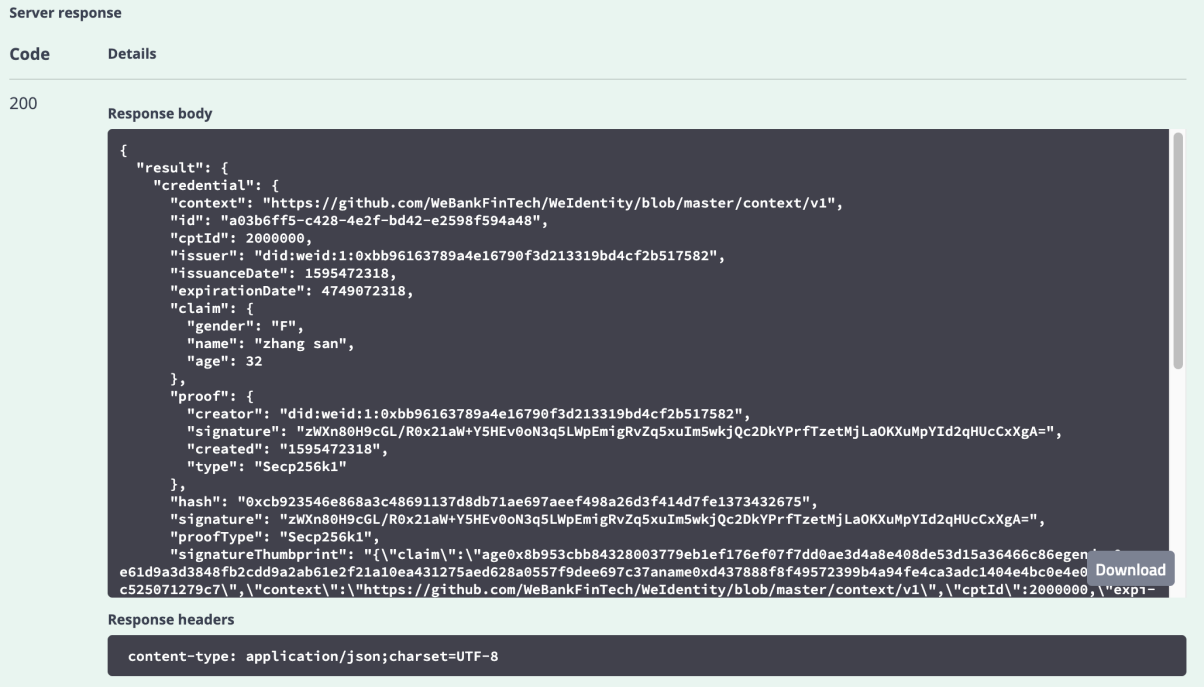
- Cpt

‘‘/step2/registCpt‘‘ publisher step1 WeID

CPT CPT ID 2000000

- Credential

‘‘/step3/createCredential‘‘ ‘‘claimData‘‘ issuer step1 WeID cptId step2 Cpt ID



Credential Credential credential

- Credential

‘‘/step1/verifyCredential‘‘ ‘‘credential‘‘

Credential

weid-sample weid-sample `com.webank.weid.demo.server.SampleApp` Java

- CPT
- WeIdentity
-
- WeIdentity
- WeIdentity
-
- WeIdentity
- WeIdentity Java SDK JDK
- WeIdentity

FAQ

- JAVA SDK FAQ

- “ ”

1.
2.
3.
4.

- **KYC**

WeIdentity KYC

1.
2. WeIdentity DID WeIdentity DID 1

WeIdentity KYC WeIdentity

- **WeIdentity**

WeIdentity WeIdentity

1.
2.

WeIdentity

- **WeIdentity “ / / / / ”**

WeIdentity

-

1. User Agent
2. User Agent

- /

User Agent

1.
2.

- “ ”

WeIdentity “ ” WeIdentity

-

- **WeIdentity DID**

1. WeIdentity SDK json WeIdentity Document User Agent ;
2. WeIdentity SDK WeIdentity WeIdentity Document WeIdentity Document WeIdentity A B A B A B

- **Credential**

WeIdentity Credential W3C Verifiable Credential WeIdentity “Credential ” “Credential ”

-

- **WeIdentity DID DID**

WeIdentity DID W3C DID WeIdentity W3C DID WeIdentity DID

- **CPT CPT**

CPT Claim Protocol Type Credential Credential Credential Credential CPT

-

A X B M A Credential CPT101 CPT

B A B Credential B Credential

-

WeIdentity DID Recovery WeIdentity Recovery WeIdentity DID N

- **Credential**

WeIdentity Credential Credential Credential Credential ID Credential Credential

---

- **Credential**

Credential ECDSA RSA ECDSA RSA

  - how-big-an-rsa-key-is-considered-secure-today
  - how-much-stronger-is-rsa-2048-compared-to-rsa-1024
  - 2048-bit RSA , 2030 Asymmetric_algorithm_key_lengths
  - RSA 768 bit
  - ECDSA vs RSA (Conclusion)

  - ECDSA RSA Shor RSA key length vs. Shor's algorithm Quantum_computing_attacks

---

- **FISCO-BCOS WeIdentity**

WeIdentity

- **PDF**

NotoSansCJKtc-Regular.ttf

CentOS

1.

```
sudo mkdir -p /usr/share/fonts/chinese
sudo cp ./NotoSansCJKtc-Regular.ttf  /usr/share/fonts/chinese
```

2.

```
sudo yum -y install fontconfig ttmkfdir mkfontscale
```

3.

```
sudo mkfontscale&&
sudo mkfontdir&&
sudo fc-cache -fv
```

4.

```
fc-list
```

Ubuntu

1.

```
sudo mkdir -p /usr/share/fonts/chinese
sudo cp ./NotoSansCJKtc-Regular.ttf  /usr/share/fonts/chinese
```

2.

```
sudo apt install xfonts-utils -y
```

3.

```
sudo mkfontscale
sudo mkfontdir
sudo fc-cache -fv
```

4.

```
fc-list
```

Window

1. window10

-

:

```
Exception in thread "main" java.lang.IncompatibleClassChangeError: class com.github.fge.
↪jackson.JsonNumEquals has interface com.google.common.base.Equivalence as super class
at java.lang.ClassLoader.defineClass1(Native Method)
at java.lang.ClassLoader.defineClass(ClassLoader.java:763)
at java.security.SecureClassLoader.defineClass(SecureClassLoader.java:142)
at java.net.URLClassLoader.defineClass(URLClassLoader.java:467)
at java.net.URLClassLoader.access$100(URLClassLoader.java:73)
at java.net.URLClassLoader$1.run(URLClassLoader.java:368)
at java.net.URLClassLoader$1.run(URLClassLoader.java:362)
at java.security.AccessController.doPrivileged(Native Method)
at java.net.URLClassLoader.findClass(URLClassLoader.java:361)
at java.lang.ClassLoader.loadClass(ClassLoader.java:424)
at sun.misc.Launcher$AppClassLoader.loadClass(Launcher.java:331)
at java.lang.ClassLoader.loadClass(ClassLoader.java:357)
at com.github.fge.jsonschema.core.keyword.syntax.checkers.common.EnumSyntaxChecker.<clinit>
↪(EnumSyntaxChecker.java:46)
at com.github.fge.jsonschema.core.keyword.syntax.dictionaries.CommonSyntaxCheckerDictionary.
↪<clinit>(CommonSyntaxCheckerDictionary.java:152)
at com.github.fge.jsonschema.core.keyword.syntax.dictionaries.DraftV3SyntaxCheckerDictionary.
↪<clinit>(DraftV3SyntaxCheckerDictionary.java:55)
```

(continues on next page)

( )

```
at com.github.fge.jsonschema.library.DraftV3Library.<clinit>(DraftV3Library.java:32)
at com.github.fge.jsonschema.cfg.ValidationConfigurationBuilder.<clinit>
↪(ValidationConfigurationBuilder.java:63)
at com.github.fge.jsonschema.cfg.ValidationConfiguration.newBuilder(ValidationConfiguration.
↪java:92)
at com.github.fge.jsonschema.cfg.ValidationConfiguration.byDefault(ValidationConfiguration.
↪java:102)
at com.github.fge.jsonschema.main.JsonSchemaFactoryBuilder.<init>(JsonSchemaFactoryBuilder.
↪java:68)
at com.github.fge.jsonschema.main.JsonSchemaFactory.newBuilder(JsonSchemaFactory.java:123)
at com.github.fge.jsonschema.main.JsonSchemaFactory.byDefault(JsonSchemaFactory.java:113)
at com.webank.weid.util.DataToolUtils.isValidJsonSchema(DataToolUtils.java:451)
at com.webank.weid.util.DataToolUtils.isCptJsonSchemaValid(DataToolUtils.java:465)
at com.webank.weid.service.impl.CptServiceImpl.validateCptJsonSchemaMap(CptServiceImpl.
↪java:358)
at com.webank.weid.service.impl.CptServiceImpl.validateCptArgs(CptServiceImpl.java:325)
at com.webank.weid.service.impl.CptServiceImpl.registerCpt(CptServiceImpl.java:167)
at Issuer.main(Issuer.java:49)
```

```
IDE      Equivalence  Idea   Ctrl+N,  Equivalence
           guava jar   pom.xml
   maven  pom.xml
```

- **FISCO-BCOS WeIdentity WeIdentity**

WeIdentity

- **Evidence**

Evidence key hash WeIdentity SDK sha3

-

```
"hash": "0x64e604787cbf194841e7b68d7cd28786f6c9a0a3ab9f8b0a0e87cb4387ab0107"

//hash  extraKey Evidence key Evidence value

{
"signer": ["did:weid:1000:0x4d3091830e74235a9c2e2041700c162ff75cc13d"],
"logs": [
    "tempLog",
    "tempLog",
    "tempLog",
    "tempLog",
    "tempLog",
    "tempLog"
],
"signature": ["AELc1QRvC+OEwwIjzZ6KrffiHpTFoxanq29H6KO3juV1NKg5Ip59/c/
↪8pgwISVNEV8mXaqhYVf2o\b0JuyZCc0f5Q="],
"updated": "1590136873"
}
```

- **CredentialService CredentialPojoService**

1. CredentialService CredentialPojoService CredentialPojoService CredentialService CredentialPojoSe
2. CredentialPojo Lite Presentation PDF/

3. AmopService requestIssueCredential Credential CredentialPojoService

- **MYSQL**

MYSQL maxActive + SDK maxActive

WeIdentity RestService

WeIdentity RestService WeIdentity RestService Java FISCO-BCOS HTTP/HTTPS WeIdentity

RestService

RestService Server

WeIdentity RestService

1. Server

1.1

Server WeIdentity-Java-SDK fisco-solc

CentOS/Ubuntu	7.2 / 16.04 64	RestServer
JDK	1.8+	1.8u141
FISCO-BCOS	1.3.8 1.2.5 2.x	Server telnet channelPort
Gradle	4.6+	4.x 5.x Gradle
MySQL	5 +	MySQL

1.2

GitHub RestService `/dist`

```
$ git clone https://github.com/WeBankFinTech/weid-http-service.git
$ cd weid-http-service
$ gradle build -x test
$ cd dist
```

develop

```
dist
  app:  jar
  lib:
  conf:
  keys/priv:
  server_status.sh
  start.sh  RestServer
  stop.sh  RestServer
```

1.3

- WeIdentity FISCO-BCOS 2.x FISCO-BCOS
- ca.crt SDK node.crt node.key dist/conf
- WeIdentity fisco.properties weidentity.properties dist/conf dist/conf/fisco.properties.tpl dist/conf/weidentity.properties.tpl .tpl

  dist/conf/fisco.properties cns.contract.follow

```
cns.contract.follow=0x161bcbd5afbdd2bb2c7f6cc31ed5897f041271c8c984284239370c1572e8545d
```

  dist/conf/weidentity.properties nodes IP channel ","

```
nodes=127.0.0.1:8812,127.0.0.1:8900
```

- WeIdentity keys/priv ecdsa_key WeIdentity output/admin/
- dist/conf/application.properties RestServer HTTP/HTTPS RESTful 6001

```
#   HTTP/HTTPS
server.port=6001
#   HTTPS  HTTP
server.http.port=6000
```

- HTTPS dist/conf/application.properties SSL Rest Service Self-Signed keystore

```
#   HTTPS      true
server.ssl.enabled=true
#   keystore
server.ssl.key-store=classpath:tomcat.keystore
# keystore         123456
server.ssl.key-store-password=
# keystore   JKS PKCS12
server.ssl.keyStoreType=JKS
# key
server.ssl.keyAlias=tomcat
```

: Rest Service HTTP/HTTPS IP 6001 HTTPS HTTP

: Postman HTTPS Postman File -> Setting -> General SSL certificate verification CA Postman File -> Setting -> Certificates

: https://hutter.io/2016/02/09/java-create-self-signed-ssl-certificates-for-tomcat/ CA

- dist/conf/application.properties WeIdentity RestService API

```
#       admin
default.passphrase=admin
```

- MySQL "dist/conf/weidentity.properties" datasource MySQL

2. Server

2.1 Server /

dist Rest Server

```
#
$ chmod +x *.sh
#
$ ./start.sh
#
$ ./server_status.sh
#
$ ./stop.sh
```

./start.sh RestServer

```
============================================================
Starting com.webank.weid.http.Application ... [SUCCESS]
============================================================
```

./server_status.sh RestServer

```
============================================================
com.webank.weid.http.Application is running(PID=100891)
============================================================
```

./stop.sh RestServer

```
============================================================
Stopping com.webank.weid.http.Application ... [SUCCESS]
============================================================
```

3. Postman RestServer API

RestServer HTTP/HTTPS RESTful API API Postman

- Postman Import `weidentity-restservice.postman_environment.json` `invoke.postman_collection.json` GitHub
- **`weidentity-restservice` `host` `httpport`**
  - `host` `RestServer`
  - `httpport` Server
- Invoke
- Invoke API CreateWeId API WeIdentity DID

RestService

API HTTP/HTTPS WeIdentity

WeIdentity RestService API

RestService API

1.

API json

```
{
    "functionArg":  SDK   json  {
        ...
    },
    "transactionArg":  json  {
        "invokerWeId":    WeIdentity DID
    }
    "functionName":  SDK
    "v": API
}
```

- functionArg SDK SDK
- **transactionArg invokerWeId WeIdentity DID**
  – CreateAuthorityIssuer
- functionName SDK WeIdentity Java SDK
- v API

API json

```
{
    "respBody":  SDK    json  {
    }
    "ErrorCode":
    "ErrorMessage":     "success"
}
```

result SDK

2. WeIdentity DID

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	createWeId	Y
functionArg		Y
transactionArg		Y
v		Y

: RestService WeID WeID RestService WeID

```
{
    "functionArg": {
    },
    "transactionArg": {
    },
    "functionName": "createWeId",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	WeIdentity DID

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
}
```

3. WeIdentity DID Document

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	getWeIdDocument	Y
functionArg		Y
functionArg.weId	WeIdentity DID SDK	Y
transactionArg		N
v		Y

```
{
    "functionArg": {
        "weId": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
    },
    "transactionArg": {
    },
    "functionName": "getWeIdDocument",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	WeIdentity DID Document

```
{
    "respBody": {
        "@context" : "https://w3id.org/did/v1",
        "id" : "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
        "created" : 1553224394993,
        "updated" : 1553224394993,
        "publicKey" : [ ],
        "authentication" : [ ],
        "service" : [ ]
    },
    "ErrorCode": 0,
    "ErrorMessage": "success"
}
```

4. AuthorityIssuer

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	registerAuthorityIssuer	Y
functionArg		Y
functionArg.weId	WeIdentity DID SDK	Y
functionArg.name		Y
transactionArg		Y
transactionArg.invokerWeId	WeIdentity DID application.properties	Y
v		Y

```
{
    "functionArg": {
        "weid": "did:weid:0x1Ae5b88d37327830307ab8da0ec5D8E8692A35D3",
        "name": "Sample College"
    },
    "transactionArg": {
        "invokerWeId": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
    },
    "functionName": "registerAuthorityIssuer",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	True/False

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": True
}
```

5. AuthorityIssuer

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	queryAuthorityIssuer	Y
functionArg		Y
functionArg.weId	WeIdentity DID SDK	Y
transactionArg		N
v		Y

```
{
    "functionArg": {
        "weId": "did:weid:0x1ae5b88d37327830307ab8da0ec5d8e8692a35d3"
    },
    "transactionArg": {
    },
    "functionName": "queryAuthorityIssuer",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	Authority Issuer

```
{
    "respBody": {
        "accValue": ,
```

(continues on next page)

( )

```
        "created": 16845611984115,
        "name": "Sample College",
        "weid": "did:weid:0x1ae5b88d37327830307ab8da0ec5d8e8692a35d3"
    }
    "ErrorCode": 0
    "ErrorMessage": "success"
}
```

6. CPT

	weid/api/invoke
Method	POST
Content-Type	application/json

:

Key	Value	Required
functionName	registerCpt	Y
functionArg		Y
functionArg.cptJsonSchema	CPT Json Schema SDK	Y
functionArg.weId	CPT	Y
transactionArg		Y
transactionArg.invokerWeId	WeIdentity DID	Y
v		Y

```
CPT Json Schema

 Json Schema    Json    Json     CPT
WeIdentity   http://json-schema.org/draft-04/schema#
```

```
{
  "functionArg": {
      "weId": "did:weid:0x1ae5b88d37327830307ab8da0ec5d8e8692a35d3",
      "cptJsonSchema":{
          "title": "cpt",
          "description": "this is cpt",
          "properties": {
              "name": {
                  "type": "string",
                  "description": "the name of certificate owner"
              },
              "gender": {
                  "enum": [
                      "F",
                      "M"
                  ],
                  "type": "string",
                  "description": "the gender of certificate owner"
              },
              "age": {
                  "type": "number",
```

(continues on next page)

( )

```
                "description": "the age of certificate owner"
            }
        },
        "required": [
            "name",
            "age"
        ]
    }
  },
  "transactionArg": {
      "invokerWeId": "did:weid:0x1ae5b88d37327830307ab8da0ec5d8e8692a35d3"
  }
  "functionName": "registerCpt"
  "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	cptBaseInfo

```
{
    "respBody": {
        "cptId": 2000001,
        "cptVersion": 1
    },
    "ErrorCode": 0,
    "ErrorMessage": "success"
}
```

7. CPT

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	queryCpt	Y
functionArg		Y
functionArg.cptId	CPT ID SDK	Y
transactionArg		N
v		Y

```
{
    "functionArg": {
        "cptId": 10,
    },
    "transactionArg": {
    },
    "functionName": "queryCpt",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	CPT

```
{
    "respBody": {
        "cptBaseInfo" : {
            "cptId" : 10,
            "cptVersion" : 1
        },
        "cptId" : 10,
        "cptJsonSchema" : {
            "$schema" : "http://json-schema.org/draft-04/schema#",
            "title" : "a CPT schema",
            "type" : "object"
        },
        "cptPublisher" : "did:weid:0x104a58c272e8ebde0c29083552ebe78581322908",
        "cptSignature" : "HJPbDmoi39xgZBGi/
↪aj1zB6VQL5QLyt4qTV6GOvQwzfgUJEZTazKZXe1dRg5aCt8Q44GwNF2k+l1rfhpY1hc/ls=",
        "cptVersion" : 1,
        "created" : 1553503354555,
        "metaData" : {
            "cptPublisher" : "did:weid:0x104a58c272e8ebde0c29083552ebe78581322908",
            "cptSignature" : "HJPbDmoi39xgZBGi/
↪aj1zB6VQL5QLyt4qTV6GOvQwzfgUJEZTazKZXe1dRg5aCt8Q44GwNF2k+l1rfhpY1hc/ls=",
            "created" : 1553503354555,
            "updated" : 0
        },
        "updated" : 0
    },
    "ErrorCode": 0,
    "ErrorMessage": "success"
}
```

8. Credential

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	createCredential	Y
functionArg		Y
functionArg.claim	claim Json SDK	Y
functionArg.cptId	CPT ID	Y
functionArg.issuer	issuer WeIdentity DID	Y
functionArg.expirationDate	UTC	Y
transactionArg		Y
transactionArg.invokerWeId	WeIdentity DID	Y
v		Y

Json signature

```
{
    "functionArg": {
        "cptId": 10,
        "issuer": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
        "expirationDate": "2019-04-18T21:12:33Z",
        "claim": {
            "name": "zhang san",
            "gender": "F",
            "age": 18
        },
    },
    "transactionArg": {
        "invokerWeId": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
    },
    "functionName": "createCredential",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	Credential

:

```
{
  "respBody": {
      "@context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
      "claim": {
          "content": "b1016358-cf72-42be-9f4b-a18fca610fca",
          "receiver": "did:weid:101:0x7ed16eca3b0737227bc986dd0f2851f644cf4754",
          "weid": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
      },
      "cptId": 2000156,
      "expirationDate": "2100-04-18T21:12:33Z",
      "id": "da6fbdbb-b5fa-4fbe-8b0c-8659da2d181b",
      "issuanceDate": "2020-02-06T22:24:00Z",
      "issuer": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
      "proof": {
          "created": "1580999040000",
          "creator": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
          "signature":
↪"G0XzzLY+MqUAo3xXkS3lxVsgFLnTtvdXM24p+G5hSNNMSIa5vAXYXXKl+Y79CO2ho5DIGPPvSs2hvAixmfIJGbw=",
          "type": "Secp256k1"
```

(continues on next page)

( )


```
        }
    },
    "errorCode": 0,
    "errorMessage": "success"
}
```


9. Credential

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	verifyCredential	Y
functionArg		Y
functionArg.claim	claim Json SDK	Y
functionArg.cptId	CPT ID	Y
functionArg.context	context	Y
functionArg.uuid	Credential UUID	Y
functionArg.issuer	issuer WeIdentity DID	Y
functionArg.issuranceDate		Y
functionArg.expirationDate		Y
functionArg.proof	Credential	Y
transactionArg		N
v		Y


```
{
    "functionArg": {
      "@context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
      "claim": {
          "content": "b1016358-cf72-42be-9f4b-a18fca610fca",
          "receiver": "did:weid:101:0x7ed16eca3b0737227bc986dd0f2851f644cf4754",
          "weid": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
      },
      "cptId": 2000156,
      "expirationDate": "2100-04-18T21:12:33Z",
      "id": "da6fbdbb-b5fa-4fbe-8b0c-8659da2d181b",
      "issuanceDate": "2020-02-06T22:24:00Z",
      "issuer": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
      "proof": {
          "created": "1580999040000",
          "creator": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
          "signature":
↪"G0XzzLY+MqUAo3xXkS3lxVsgFLnTtvdXM24p+G5hSNNMSIa5vAXYXXKl+Y79CO2ho5DIGPPvSs2hvAixmfIJGbw=",
          "type": "Secp256k1"
      }
    },
    "transactionArg": {
```


(continues on next page)

( )

```
    },
    "functionName": "verifyCredential"
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	True/False

```
{
    "respBody": true,
    "ErrorCode": 0,
    "ErrorMessage": "success"
}
```

10. CredentialPojo

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	createCredentialPojo	Y
functionArg		Y
functionArg.claim	claim Json SDK	Y
functionArg.cptId	CPT ID	Y
functionArg.issuer	issuer WeIdentity DID	Y
functionArg.expirationDate	UTC	Y
transactionArg		Y
transactionArg.invokerWeId	WeIdentity DID	Y
v		Y

Json signature

```
{
    "functionArg": {
        "cptId": 10,
        "issuer": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
        "expirationDate": "2019-04-18T21:12:33Z",
        "claim": {
            "name": "zhang san",
            "gender": "F",
            "age": 18
        },
```

(continues on next page)

( )

```
    },
    "transactionArg": {
        "invokerWeId": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
    },
    "functionName": "createCredentialPojo",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	CredentialPojo

:

```
{
  "respBody": {
      "cptId": 2000156,
      "issuanceDate": 1580996777,
      "context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
      "claim": {
          "content": "b1016358-cf72-42be-9f4b-a18fca610fca",
          "receiver": "did:weid:101:0x7ed16eca3b0737227bc986dd0f2851f644cf4754",
          "weid": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
      },
      "id": "21d10ab1-75fe-4733-9f1d-f0bad71b5922",
      "proof": {
          "created": 1580996777,
          "creator": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa#keys-0",
          "salt": {
              "content": "ncZ5F",
              "receiver": "L0c40",
              "weid": "I4aop"
          },
          "signatureValue":
↪"HEugP13uDVBg2G0kmmwbTkQXobsrWNqtGQJW6BoHU2Q2VQpwVhK382dArRMFN6BDq7ogozYBRC15QR8ueX5G3t8=",
          "type": "Secp256k1"
      },
      "type": [
          "VerifiableCredential",
          "hashTree"
      ],
      "issuer": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
      "expirationDate": 4111737153
  },
  "errorCode": 0,
  "errorMessage": "success"
}
```

11. CredentialPojo

	weid/api/invoke
Method	POST
Content-Type	application/json

Key	Value	Required
functionName	verifyCredentialPojo	Y
functionArg		Y
functionArg.claim	claim Json SDK	Y
functionArg.cptId	CPT ID	Y
functionArg.context	context	Y
functionArg.uuid	CredentialPojo UUID	Y
functionArg.issuer	issuer WeIdentity DID	Y
functionArg.issuranceDate		Y
functionArg.expirationDate		Y
functionArg.proof	Credential	Y
transactionArg		N
v		Y

```
{
    "functionArg": {
      "cptId": 2000156,
      "issuanceDate": 1580996777,
      "context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
      "claim": {
          "content": "b1016358-cf72-42be-9f4b-a18fca610fca",
          "receiver": "did:weid:101:0x7ed16eca3b0737227bc986dd0f2851f644cf4754",
          "weid": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
      },
      "id": "21d10ab1-75fe-4733-9f1d-f0bad71b5922",
      "proof": {
          "created": 1580996777,
          "creator": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa#keys-0",
          "salt": {
              "content": "ncZ5F",
              "receiver": "L0c40",
              "weid": "I4aop"
          },
          "signatureValue":
↪"HEugP13uDVBg2G0kmmwbTkQXobsrWNqtGQJW6BoHU2Q2VQpwVhK382dArRMFN6BDq7ogozYBRC15QR8ueX5G3t8=",
          "type": "Secp256k1"
      },
      "type": [
          "VerifiableCredential",
          "hashTree"
      ],
      "issuer": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
      "expirationDate": 4111737153
    },
    "transactionArg": {
    },
    "functionName": "verifyCredentialPojo"
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	True/False

```
{
    "respBody": true,
    "ErrorCode": 0,
    "ErrorMessage": "success"
}
```

RestService API

1.

API API json RestService ECDSA sha3

-

	weid/api/encode
Method	POST
Content-Type	application/json

Body

```
{
    "functionArg":  SDK   json  {
        ...
    },
    "transactionArg":   json  {
        "nonce":
    }
    "functionName":  SDK
    "v": API
}
```

- functionArg SDK SDK
- **transactionArg nonce RestService jar getNonce()**
  – nonce
- functionName SDK WeIdentity Java SDK
- v API

json

```
{
    "respBody": {
        "encodedTransaction":  Base64
```

(continues on next page)

( )

```
        "data":
    }
    "ErrorCode":
    "ErrorMessage":       "success"
}
```

encodedTransaction data encodeTransaction Base64 data nonce

: ECDSA WeID Java SDK Secp256k1 WeID Go ECDSA R S V R S 32 V Secp256k1 0 1

: R S V R S V 65 Base64 RestService RestService 1. R, S, V 65 Base64 WeID Go V 0 1 2. V+27, R, S 65 Base64 WeID Java SDK V 27 28

Java Secp256k1 Base64

```
//  web3sdk 2.2.2 weid-java-sdk 1.5
byte[] encodedTransaction = DataToolUtils
        .base64Decode("<encodedTransaction >".getBytes());
    SignatureData clientSignedData = Sign.getSignInterface().
↪signMessage(encodedTransactionClient, ecKeyPair);
    String base64SignedMsg = new String(
        DataToolUtils.base64Encode(TransactionEncoderUtilV2.
↪simpleSignatureSerialization(clientSignedData)));
```

-

	weid/api/transact
Method	POST
Content-Type	application/json

```
{
    "functionArg":  {
    },
    "transactionArg":    json   {
        "nonce":
        "data":
        "signedMessage":  Base64       encodedTransaction
    }
    "functionName":  SDK
    "v": API
}
```

RestService

- functionArg
- transactionArg nonce data Base64 encodedTransaction
- functionName SDK WeIdentity Java SDK
- v API

json

```
{
    "respBody":   SDK     json   {
    }
    "ErrorCode":
    "ErrorMessage":      "success"
}
```

result SDK API

2. WeIdentity DID

POST /weid/api/encode

Key	Value	Required
functionName	createWeId	Y
functionArg		Y
functionArg.publicKey	ECDSA 10	Y
transactionArg		Y
transactionArg.nonce		Y
v		Y

POST /weid/api/encode

```
{
    "functionArg": {
        "publicKey": "712679236821355231513532168231727831978932132185632517152735621683128"
    },
    "transactionArg": {
        "nonce": "147480060101130736550612130457634747950865349998942434640834385561582214603 9"
    },
    "functionName": "createWeId",
    "v": "1.0.0"
}
```

POST /weid/api/transact

```
{
    "functionArg": {},
    "transactionArg": {
        "nonce": "147480060101130736550612130457634747950865349998942434640834385561582214603
↪9",
        "data": "8098126382 56c1235b1231000e000000001231287bacf213c",
        "signedMessage":
↪"HEugP13uDVBg2GOkmmwbTkQXobsrWNqtGQJW6BoHU2Q2VQpwVhK382dArRMFN6BDq7ogozYBRC15QR8ueX5G3t8="
    },
    "functionName": "createWeId",
    "v": "1.0.0"
}
```

Key	Value
ErrorCode	0
ErrorMessage	
respBody	WeIdentity DID

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
}
```

3. Authority Issuer

POST /weid/api/encode

Key	Value	Required
functionName	registerAuthorityIssuer	Y
functionArg		Y
functionArg.name		Y
functionArg.weId	WeIdentity DID SDK	Y
transactionArg		Y
transactionArg.nonce		Y
v		Y

POST /weid/api/encode

```
{
    "functionArg": {
        "name": "BV-College",
        "weId": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
    },
    "transactionArg": {
        "nonce": "147480060101130736550612130457634747950865349998942434640834385561582214603 9"
    },
    "functionName": "registerAuthorityIssuer",
    "v": "1.0.0"
}
```

POST /weid/api/transact

```
{
    "functionArg": {},
    "transactionArg": {
        "nonce": "147480060101130736550612130457634747950865349998942434640834385561582214603
↪9",
        "data": "809812638256c1235b1231000e000000001231287bacf213c",
        "signedMessage":
↪"HEugP13uDVBg2GOkmmwbTkQXobsrWNqtGQJW6BoHU2Q2VQpwVhK382dArRMFN6BDq7ogozYBRC15QR8ueX5G3t8="
    },
    "functionName": "registerAuthorityIssuer",
    "v": "1.0.0"
}
```

Key	Value
ErrorCode	0
ErrorMessage	
respBody	Authority Isser

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": {
        "accValue": ,
        "created": "1581420650",
        "name": "BV-College",
        "weId": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
    }
}
```

4. CPT

POST /weid/api/encode

Key	Value	Required
functionName	registerCpt	Y
functionArg		Y
functionArg.cptJsonSchema	CPT Json Schema SDK	Y
functionArg.weId	CPT	Y
functionArg.cptSignature	cptJsonSchema	Y
transactionArg		Y
transactionArg.nonce		Y
v		Y

POST /weid/api/encode

```
{
    "functionArg": {
        "weId": "did:weid:0x1ae5b88d37327830307ab8da0ec5d8e8692a35d3",
        "cptJsonSchema": {
            "title": "cpt",
            "description": "this is cpt",
            "properties": {
                "name": {
                    "type": "string",
                    "description": "the name of certificate owner"
                },
                "gender": {
                    "enum": [
                        "F",
                        "M"
                    ],
                    "type": "string",
                    "description": "the gender of certificate owner"
                },
                "age": {
                    "type": "number",
                    "description": "the age of certificate owner"
                }
            },
            "required": [
                "name",
                "age"
            ]
        }
        "cptSignature":
↪"BaUeP13uDVBg2G0kmmwbTkQXobsrWNqtGQJW6BoHU2Q2VQpwVhK382dArRMFN6BDq7ogozYBRC15QR8ueX5G3t8="
```

(continues on next page)

( )

```
    },
    "transactionArg": {
        "nonce": "147480060101130736550612130457634747950865349998942434640834385561582214603 9"
    },
    "functionName": "registerCpt",
    "v": "1.0.0"
}
```

POST /weid/api/transact

```
{
    "functionArg": {},
    "transactionArg": {
        "nonce": "147480060101130736550612130457634747950865349998942434640834385561582214603
↪9",
        "data": "809812638256c1235b1231000e000000001231287bacf213c",
        "signedMessage":
↪"HEugP13uDVBg2GOkmmwbTkQXobsrWNqtGQJW6BoHU2Q2VQpwVhK382dArRMFN6BDq7ogozYBRC15QR8ueX5G3t8="
    },
    "functionName": "registerAuthorityIssuer",
    "v": "1.0.0"
}
```

Key	Value
ErrorCode	0
ErrorMessage	
respBody	Authority Isser

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": {
        "cptId": 2000001,
        "cptVersion": 1
    }
}
```

5. CredentialPojo

CredentialPojo POST /weid/api/encode

POST /weid/api/encode

Key	Value	Required
functionName	createCredentialPojo	Y
functionArg		Y
functionArg.claim	claim Json SDK	Y
functionArg.cptId	CPT ID	Y
functionArg.issuer	issuer WeIdentity DID	Y
functionArg.expirationDate	UTC	Y
transactionArg		Y
v		Y

Json signature

```
{
    "functionArg": {
        "cptId": 10,
        "issuer": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
        "expirationDate": "2019-04-18T21:12:33Z",
        "claim": {
            "name": "zhang san",
            "gender": "F",
            "age": 18
        },
    },
    "transactionArg": {
    },
    "functionName": "createCredentialPojo",
    "v": "1.0.0"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	CredentialPojo

:

```
{
  "respBody": {
      "cptId": 2000156,
      "issuanceDate": 1580996777,
      "context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
      "claim": {
          "content": "b1016358-cf72-42be-9f4b-a18fca610fca",
          "receiver": "did:weid:101:0x7ed16eca3b0737227bc986dd0f2851f644cf4754",
          "weid": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa"
      },
      "id": "21d10ab1-75fe-4733-9f1d-f0bad71b5922",
      "proof": {
          "created": 1580996777,
          "creator": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa#keys-0",
          "salt": {
              "content": "ncZ5F",
              "receiver": "L0c40",
              "weid": "I4aop"
          },
          "signatureValue": "HJPbDmoi39xgZBGi/
↪aj1zB6VQL5QLyt4qTV6GOvQwzfgUJEZTazKZXe1dRg5aCt8Q44GwNF2k+l1rfhpY1hc/ls=",
          "type": "Secp256k1"
      },
      "type": [
          "VerifiableCredential",
          "hashTree"
      ],
      "issuer": "did:weid:101:0xfd28ad212a2de77fee518b4914b8579a40c601fa",
      "expirationDate": 4111737153
  },
  "errorCode": 0,
  "errorMessage": "success"
}
```

CredentialPojo proof signatureValue - base64 - byte[] secp256k1 hash - hash byte[] hash Java web3sdk SignMessage() - r s v - - byte base64 RestService

ECDSA Base64 Java Go

```
String signature = DataToolUtils.sign(new String(DataToolUtils.base64Decode(signatureValue)),␣
↪privateKey);
```

```
base64SignatureValue := credentialEncodeResponse.RespBody.Proof.SignatureValue
signatureValue, err3 := base64.StdEncoding.DecodeString(base64SignatureValue)
hashedMsg := Hash(signatureValue)
doubleHashedMsg := Hash(hashedMsg)
privateKeyBytes := ConvertPrivateKeyBigIntToPrivateKeyBytes(privateKeyBigInt)
signatureBytes, err4 := SignSignature(doubleHashedMsg, privateKeyBytes)
signatureBase64String := base64.StdEncoding.EncodeToString(signatureBytes)
```

Weldentity Endpoint Service API

1. Endpoint

	weid/api/endpoint
Method	GET
Content-Type	application/json

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": [
        {
            "requestName": "create-passphrase",
            "inAddr": [
                "127.0.0.1:6010",
                "127.0.0.1:6011"
            ],
            "description": "Create a valid random passphrase"
        },
        {
            "requestName": "verify-passphrase",
            "inAddr": [
                "127.0.0.1:6012",
                "127.0.0.1:6013"
            ],
            "description": "Verify a passphrase"
        }
    ]
}
```

2. Endpoint

	weid/api/endpoint/{endpoint}
Method	POST
Content-Type	application/json

Key	Value	Required
/{endpoint}	API API String	Y
body	‘‘‘ API	Y

```
{
    "body": "did:weid:0xfd28ad212a2de77fee518b4914b8579a40c601fa```25"
}
```

: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	SDK String

```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": "did:weid:0x1Ae5b88d37327830307ab8da0ec5D8E8692A35D3",
}
```

Weldentity API

	weid/api/authorize/fetch-data
Method	POST
Content-Type	application/json

Key	Value	Required
authToken	CPT101 DataToolUtils.serialize()	Y
signedNonce	Nonce	Y

```
{
  "authToken": {
      "claim": {
          "duration": 360000,
          "fromWeId": "did:weid:101:0x69cd071e4be5fd878e1519ff476563dc2f4c6168",
```

(continues on next page)

( )


```
          "resourceId": "4b077c17-9612-42ee-9e36-3a3d46b27e81",
          "serviceUrl": "http://127.0.0.1:6010/fetch-data",
          "toWeId": "did:weid:101:0x68bedb2cbe55b4c8e3473faa63f121c278f6dba9"
      },
      "context": "https://github.com/WeBankFinTech/WeIdentity/blob/master/context/v1",
      "cptId": 101,
      "expirationDate": 1581347039,
      "id": "48b75424-9411-4d22-b925-4e730b445a31",
      "issuanceDate": 1580987039,
      "issuer": "did:weid:101:0x69cd071e4be5fd878e1519ff476563dc2f4c6168",
      "proof": {
          "created": 1580987039,
          "creator": "did:weid:101:0x69cd071e4be5fd878e1519ff476563dc2f4c6168#keys-0",
          "salt": {
              "duration": "fmk5A",
              "fromWeId": "DEvFy",
              "resourceId": "ugVeN",
              "serviceUrl": "nVdeE",
              "toWeId": "93Z1E"
          },
          "signatureValue":
↪"HCZwyTzGst87cjCDaUEzPrO8QRlsPvCYXvRTUVBUTDKRSoGDgu4h4HLrMZ+emDacRnmQ/yke38u1jBnilNnCh6c=",
          "type": "Secp256k1"
      },
      "type": ["VerifiableCredential", "hashTree"]
  },
  "signedNonce": "123123"
}
```


: application/json

Key	Value
ErrorCode	0
ErrorMessage	
respBody	SDK String


```
{
    "ErrorCode": 0,
    "ErrorMessage": "success",
    "respBody": "sample data",
}
```


RestService

RestService GitHub RestService

WeIdentity RestService

1.

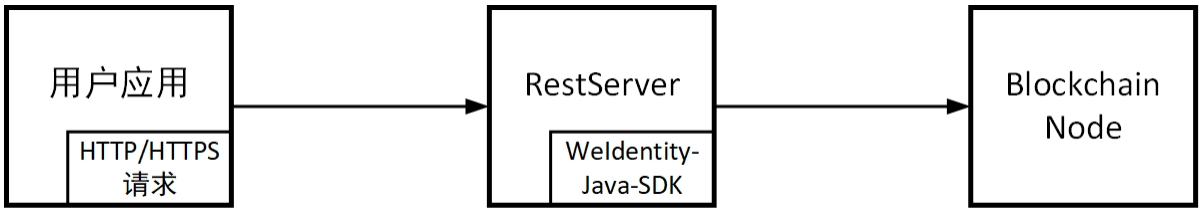
RestService

- RestService WeIdentity “ ” RestService
- Java SDK HTTP RestService SDK HTTP WeIdentity

- RestService API

2. RestService

2.1



RestService

- app HTTP
- rest-server Server
- weid-java-sdk WeIdentity SDK jar

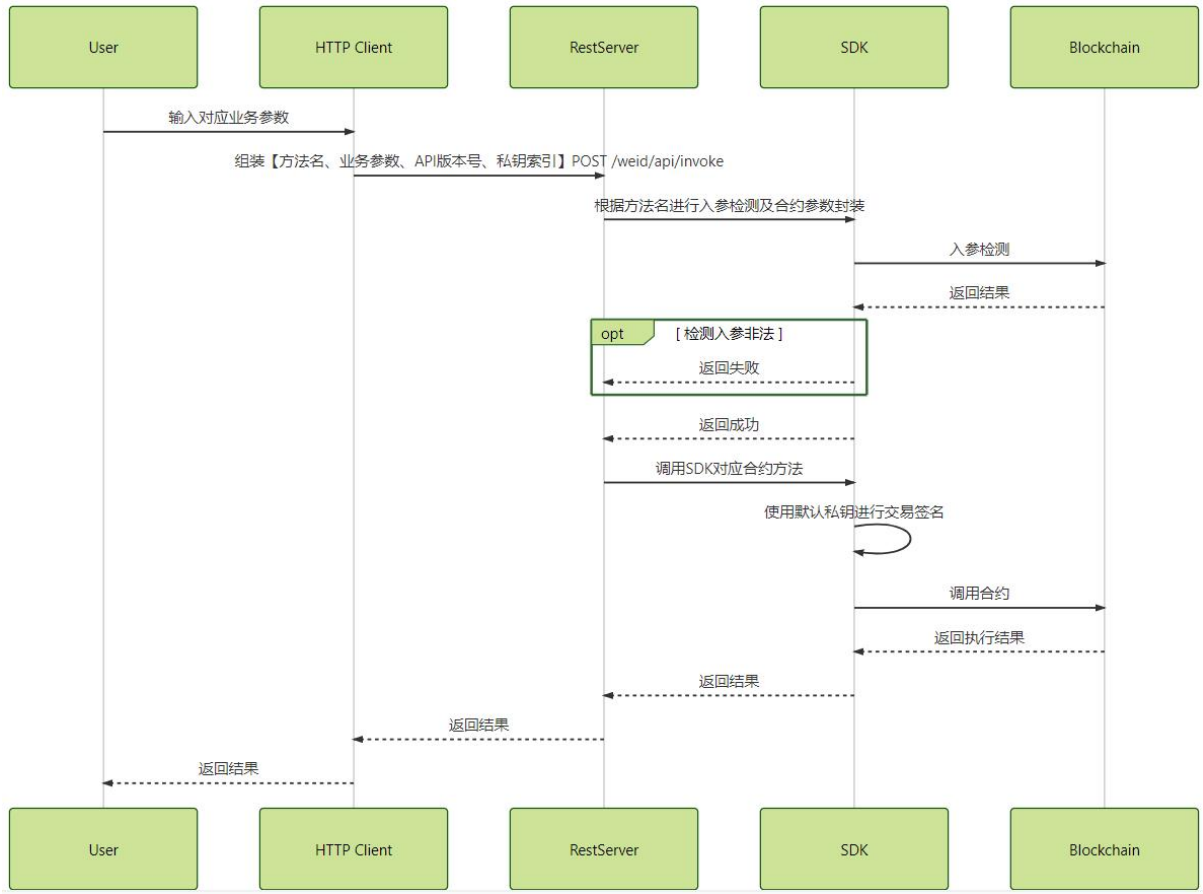
2.2

- RESTful RestService sdk

- POST /weid/api/encode RestService
- RestService
- ECDSA sha3 POST /weid/api/transact RestService
- RestService

- POST /weid/api/invoke RestService
- RestService weid-java-sdk

3.



- API POST /weid/api/invoke
- Server SDK Server SDK

WeIdentity Endpoint Service

Java HTTP/HTTPS RPC Endpoint

WeIdentity Endpoint WeIdentity RestService Java WeIdentity Java SDK Endpoint RestService API

Endpoint Service

Endpoint Service

WeIdentity Endpoint Service

RPC WeIdentity Endpoint Service RestService Java SDK

HTTP/HTTPS Endpoint Endpoint Endpoint Service

1. RestService

Endpoint Service Rest Service Rest Service

`dist/conf` `application.properties` `server.hostport.list` Rest Service IP Endpoint Service `fetch.period.seconds` Endpoint

```
#     Endpoint
fetch.period.seconds=60
#
server.hostport.list=127.0.0.1:6010,127.0.0.2:6011
```

2. Java SDK Endpoint

Rest Service Endpoint Java SDK Endpoint Java-SDK

- **Endpoint** `EndpointFunctor` `execute()` `getDescription()`
  - `execute()` RPC
  - `getDescription()`
- `registerEndpoint()` EndpointHandler Endpoint
- Endpoint Endpoint
- Endpoint
- Endpoint `RpcServer` `main()`

  EndpointSample.java

3.

Endpoint Service WeIdentity-Java-SDK Java-SDK

`src/main/resources` `weidentity.properties` `rpc.listener.port`

```
# RPC
rpc.listener.port=6010
```

Endpoint `EndpointSample.java`

```
java -cp <$your class path> com.webank.weid.suite.endpoint.EndpointSample
```

IDE RPC Server

```
Trying to receive incoming traffic at Port: 6010
```

4. Endpoint

Endpoint 60 Endpoint

REST API “WeIdentity Endpoint Service API” Endpoint

Endpoint Service

RestService API Endpoint Service API Endpoint

Endpoint Service

WeIdentity Endpoint Service Credential Endpoint Service

CPT ID 101 Claim

- fromWeId WeID Issuer
- toWeId WeID
- duration
- serviceUrl HTTP/HTTPS URL https://127.0.0.1:6010/data-auth/guest
- ID resourceId ID UUID

CPT101 ID Endpoint Service serviceUrl Endpoint Service RPC ID API

- **Endpoint Service RPC** **smart-socket 1.4.2** **Java AIO**
  - Endpoint Service RPC
  - Endpoint Service
- Endpoint Service 1:N Endpoint Endpoint

WeIdentity

WeIdentity

WeIdentity Solidity Solidity Booleans Integers Address Bytes Enum) Struct Mapping Array BCOS

WeIdentity

- **WeIdentity DID** ID DID Distributed IDentity DID Document DID
- **WeIdentity Authority** DID

WeIdentity DID

DID DID Document DID Document

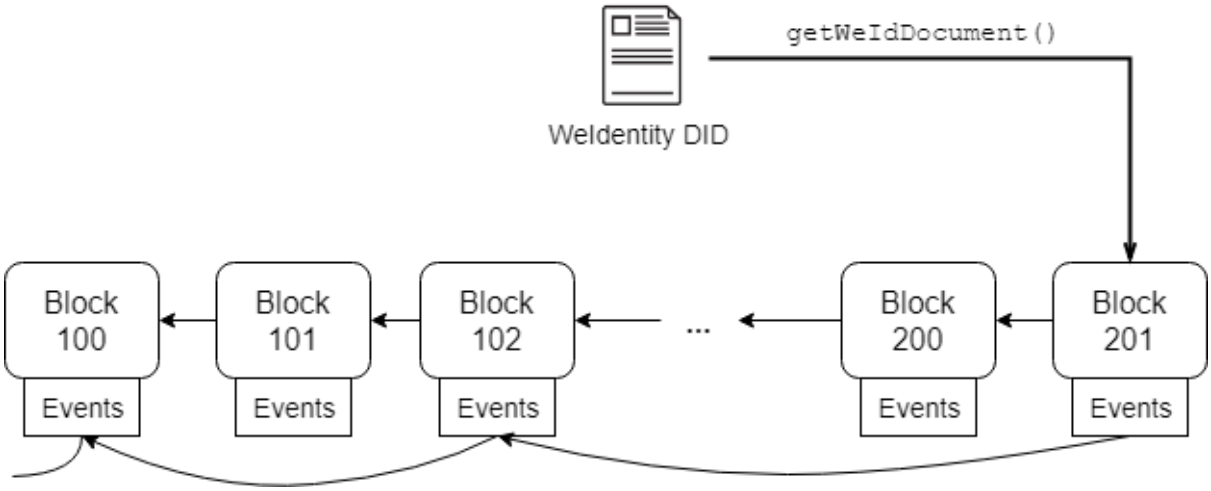
- DID
- DID Document

WeIdentity Linked Event

Linked Event Solidity Event DID Document Solidity Event Event log DID Event DID Document Event

- DID DID
- DID
- DID

- DID Document
- Event
- DID Document Event Document



Linked Event

- Solidity Event DID Document
-
- O(N) Document WeIdentity DID WeIdentity

WeIdentity Authority

Authority WeIdentity

- DID

Authority Issuer CPT DID Authority Issuer Committee Authority Issuer WeIdentity —

-

WeIdentity

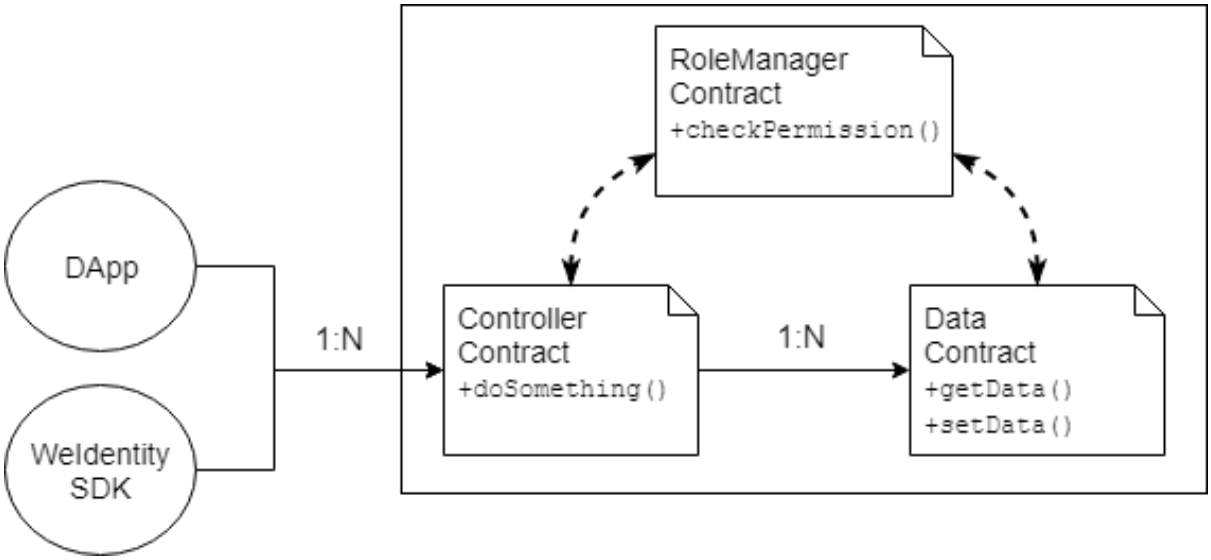
ds-auth OpenZepplin Role WeIdentity Authority

WeIdentity

- DID WeIdentity ID
- Authority Issuer CPT
- Committee Member Authority Issuer
- Administrator Committee Member Authority Issuer

WeIdentity

-
-
-



WeIdentity

- SDK DApp SDK
-

WeIdentity RoleManager WeIdentity

-
- checkPermission()
-
- WeIdentity checkPermission()

- WeIdentity

WeIdentity

- WeIdentity ds-auth OpenZepplin Role
- tx.origin msg.sender DID msg.sender WeIdentity DID DID

Specific Issuer Issuer

WeIdentity Authority Issuer Specific Issuer Authority Issuer

WeIdentity Evidence

WeIdentity DID + Evidence Hash WeIdentity SDK 1.5.2+ Evidence WeID Linked-Event hash log log

- Hash
- createEvidence
- addLog
- log
-
-

WeIdentity CPT

WeIdentity CPT Claim Protocol Type Claim CPT - CPT jsonSchema jsonSchema Clai CPT ID 1~1000 CPT 1000~2000000 CPT 2000000 CPT

CPT

CPT ID 1~1000 WeIdentity CPT WeIdentity WeIdentity CPT CPT CPT

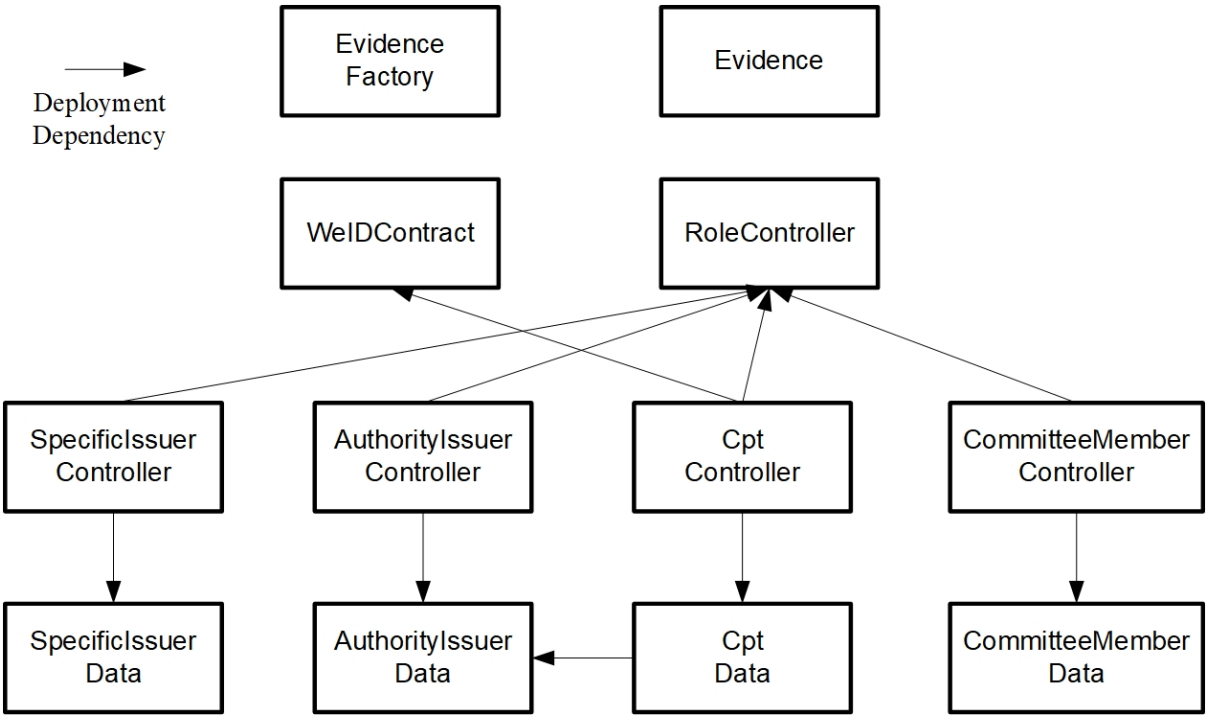
CPT

CPT ID 1000~2000000 Authority CPT Authority Issuer

CPT

CPT ID 2000000 WeID CPT

WeIdentity


Deployment
Dependency
Evidence Factory
Evidence
WeIDContract
RoleController
SpecificIssuer Controller
AuthorityIssuer Controller
Cpt Controller
CommitteeMember Controller
SpecificIssuer Data
AuthorityIssuer Data
Cpt Data
CommitteeMember Data