



Credentials Bootcamp

Learning Impact
June, 2023



Agenda

10:30 - 12:00 / 15:00 - 16:30

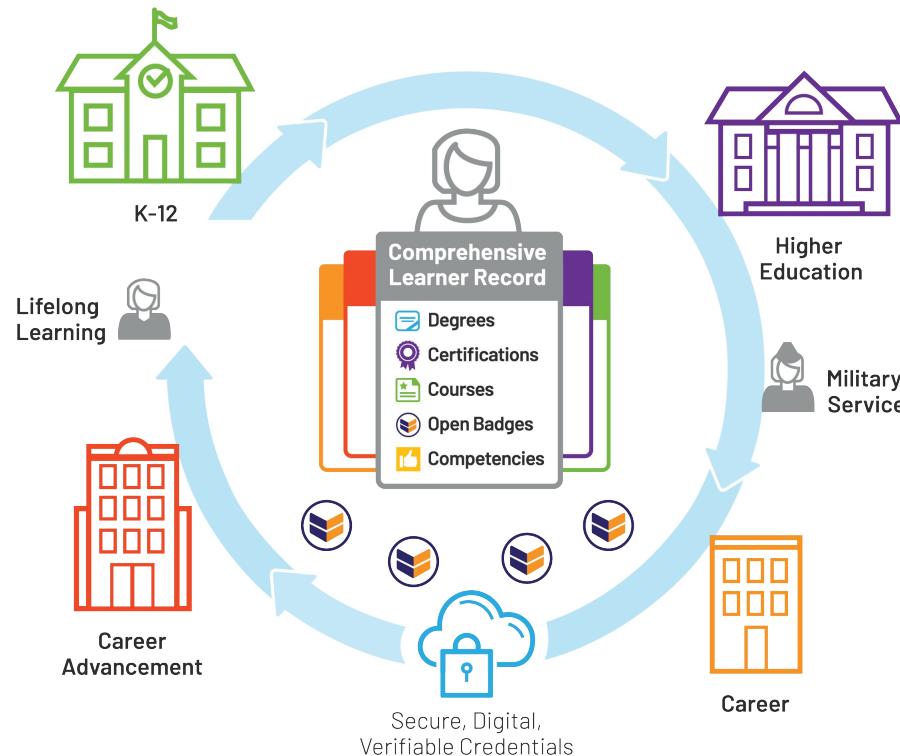
1. What problems do these new specifications solve?
2. Detailed tour of each of the specifications
3. How to leverage the tight integration between the two specifications
4. How to leverage the alignment with the W3C Verifiable Credentials standard
5. How to migrate from OB 2.0/2.1 and CLR 1.0
6. Conformance and certification expectations and how to use the Conformance Test Systems
7. The relationship between OB and CLR and the broader 1EdTech Ecosystem including how to integrate with IMS CASE.

Introduction

What problems do these new specifications solve?



Digital Credentials Ecosystem



Open Standards for Learning and Employment Records

Open Badges



Token Representing an Achievement

Comprehensive Learner Record (CLR)



Longitudinal Record of Achievements and Milestones

CASE



Publish and Relate Skills

1Edtech Open Standards Enable an Interoperable Ecosystem



Compatible with Emerging **Digital Wallet** Standard

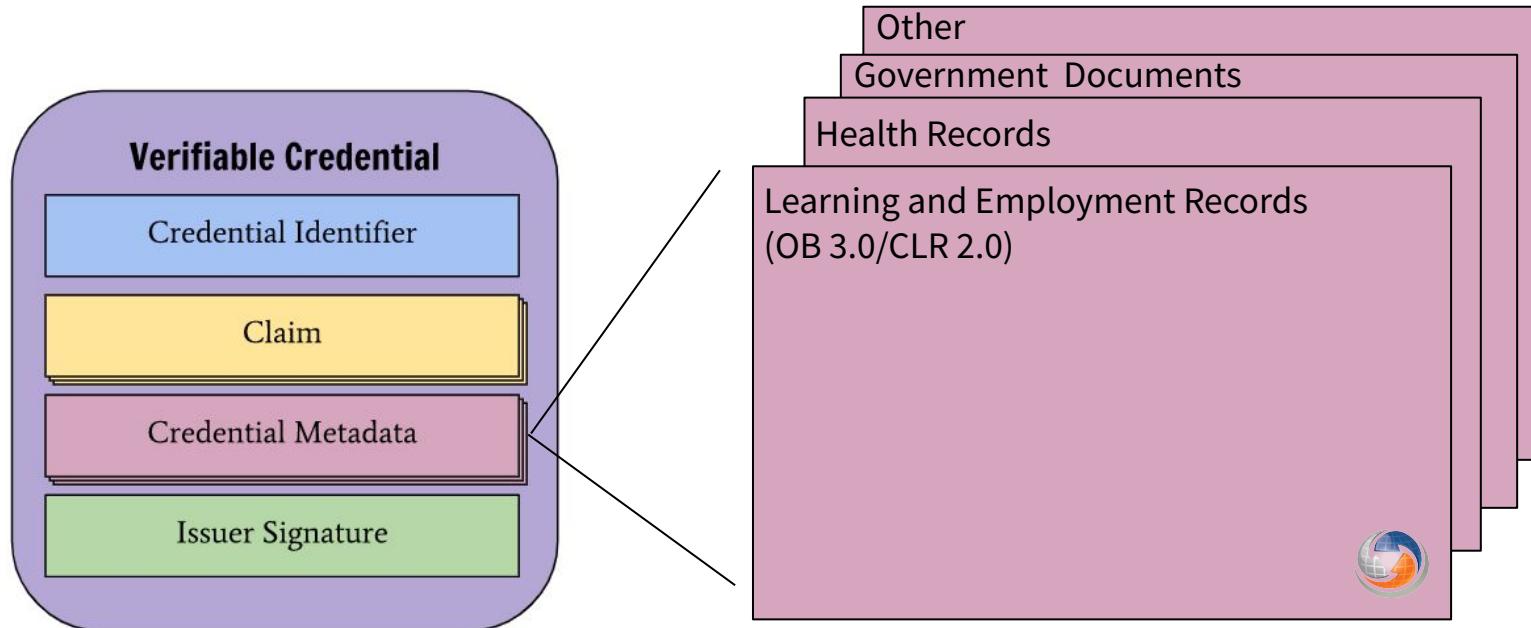


Figure 5 The basic components of a verifiable credential.



Learner Sovereignty

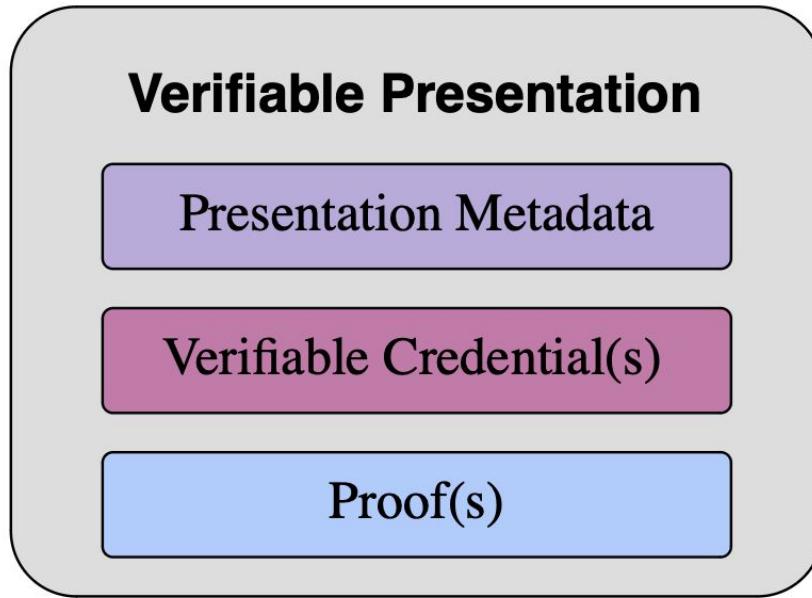


Figure 7 Basic components of a verifiable presentation.

[Copyright](#) © 2018 [W3C®](#) ([MIT](#), [ERCIM](#), [Keio](#), [Beihang](#)). W3C [liability](#), [trademark](#) and [permissive document license](#) rules apply.

Learner Mobility



OB 3.0 & CLR 2.0

Detailed tour of each of the specifications



Specifications

Candidate Public Final



Open Badges 3.0

<http://www.imsglobal.org/spec/ob/v3p0>

Power Learner Potential



Comprehensive Learner Record 2.0

<https://www.imsglobal.org/spec/clr/v2p0>

@LearningImpact | IEDTECH.org

© 2022 1EdTech Consortium, Inc. All rights reserved.



W3C's Verifiable Credentials

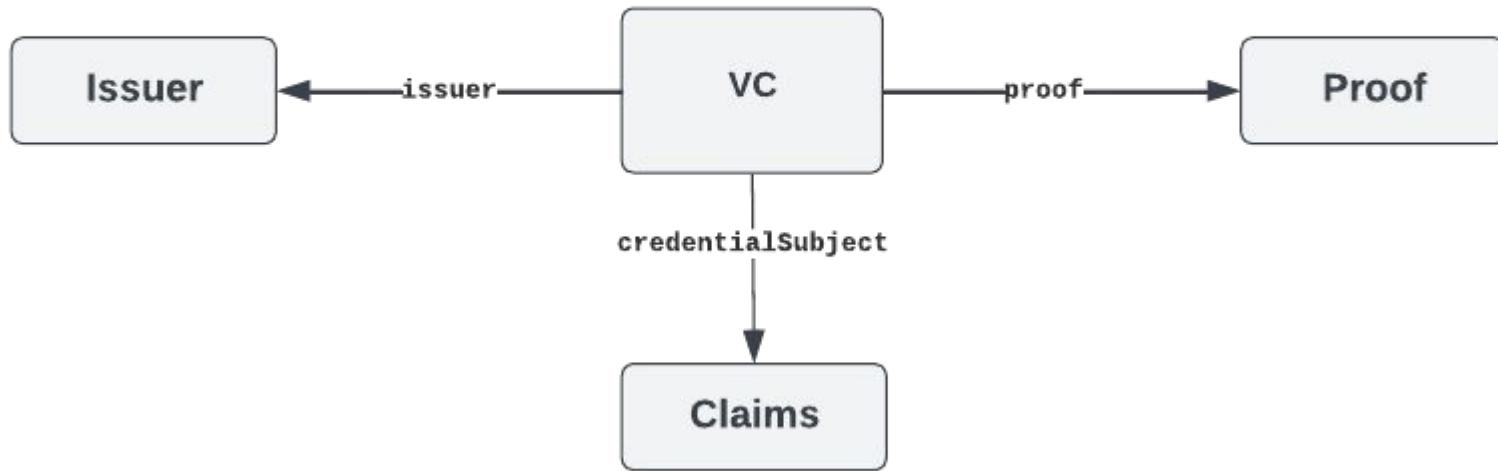
OB 3.0 and CLR 2.0 are a **profile** of VC:

- Follow VC's Data Model v1.1
- Add additional constraints to the model



<https://www.w3.org/TR/vc-data-model/>

W3C's Verifiable Credentials



W3C's Verifiable Credentials

Issuer

- A entity who asserts claims about a subject.
- *issuer* property.
- URI or object containing an *id* property.
- [Recommended] URI / id referencing a machine-readable document for verification.
 - HTTP
 - DID
 - ...



W3C's Verifiable Credentials

Claim

- A statement about a subject.
- Expressed using properties about the subject.
- *credentialSubject* property.
- MAY contain an *id*.

W3C's Verifiable Credentials

Proof

- Cryptographic mechanism.
- Used to detect tampering and verify authorship.
- A credential may include multiple proofs.
- **External** (JSON Web Token) or **Embedded**.
- (Embedded) *proof* property.
- (Embedded) signature, signing entity, and signing date.

W3C's Verifiable Credentials

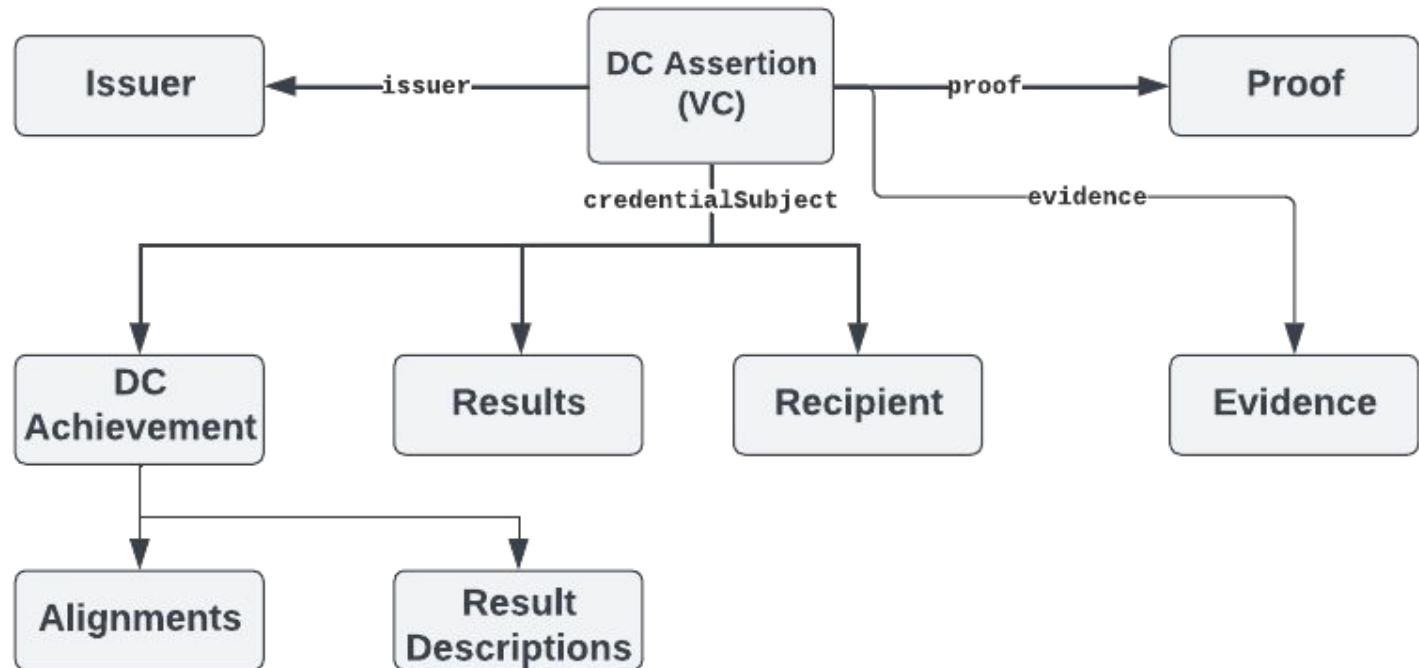
Metadata

- Context. MUST contain <https://www.w3.org/2018/credentials/v1>
- Type. MUST contain VerifiableCredential. (type)
- Issuance date. (issuanceDate)
- Expiration date. (expirationDate)
- Status. (credentialStatus)
- Evidences. (evidence)
- Data schemas (credentialSchema)
- Refreshing (refreshService)



Open Badges 3.0

AchievementCredential



Open Badges 3.0

AchievementCredential

- Context. MUST contain
<https://purl.imsglobal.org/spec/ob/v3p0/context-3.0.1.json>
- Type. MUST contain AchievementCredential or
OpenBadgeCredential. (type)
- Endorsements (JWT + embedded)

Open Badges 3.0

AchievementSubject

- Type. MUST contain AchievementSubject. (*type*)
- Achievement. (*achievement*)
- Recipient. (*id / identifier*)
- Baked Image. (*image*)
- Narrative. (*evidence*)
- Entity which assesses the achievement (*source*)

Open Badges 3.0

AchievementSubject

- Dates. (*activityStartDate*, *activityEndDate*)
- Credits. (*creditsEarned*)
- Licenses. (*licenseNumber*)
- Role. (*role*)
- Term. (*term*)
- Results: Rubrics, status. (*results*)

Open Badges 3.0

Achievement

- Type. (*achievementType*)
- Who created the achievement definition. (*creator*)
- Criteria describing how to earn the achievement. (*criteria*)
- Related achievements. (*related*)
- Possible results. Rubrics. (*resultDescription*)
- Endorsements (*endorsement* / *endorsementJwt*)

Open Badges 3.0

Achievement

- Name and description. (*name, description*)
- Category, discipline, area of study. (*fieldOfStudy, specialization*)
- Credits. (*creditsAvailable*)
- Keywords. (*tag*)
- Image (*image*)
- Version. (*version*)
- Objectives or educational standards. (*alignment*)

Open Badges 3.0

AchievementType

- Achievement
- ApprenticeshipCertificate
- Assessment
- Assignment
- AssociateDegree
- Award
- Badge
- BachelorDegree
- Certificate
- CertificateOfCompletion
- Certification
- CommunityService
- Competency
- Course
- CoCurricular
- Degree
- Diploma
- DoctoralDegree
- Fieldwork
- GeneralEducationDevelopment
- JourneymanCertificate
- LearningProgram
- License
- Membership
- ProfessionalDoctorate
- QualityAssuranceCredential
- MasterCertificate
- MasterDegree
- MicroCredential
- ResearchDoctorate
- SecondarySchoolDiploma
- ...



Open Badges 3.0

Alignment

- Framework (*targetFramework*)
- Type (*targetType*)
- URL (*targetUrl*)
- Name (*targetName*)
- Description (*targetDescription*)
- Code (*targetCode*)

Open Badges 3.0

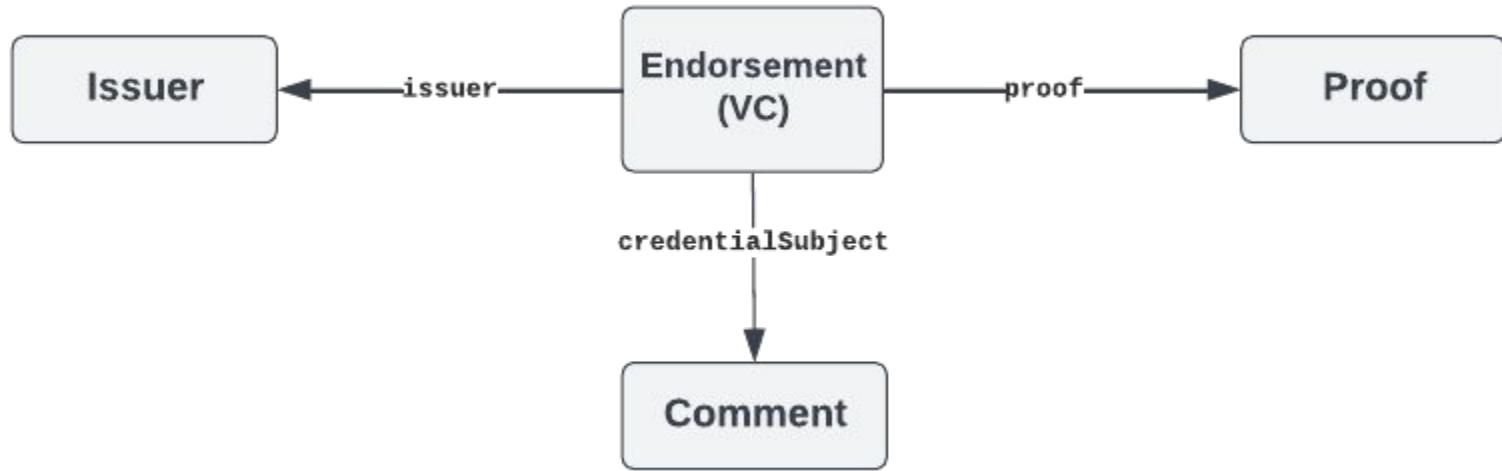
AlignmentTargetType

- CTDL-ASN/CTDL competency published by Credential Engine. (`ceasn:Competency`)
- CTDL Credential published by Credential Engine. (`ceterms:Credential`)
- CASE Framework Item. (`CFItem`)
- CASE Framework Rubric. (`CFRubric`)
- CASE Framework Rubric Criterion. (`CFRubricCriterion`)
- CASE Framework Rubric Criterion Level. (`CFRubricCriterionLevel`)
- Credential Engine Item. (`CTDL`)
- ...



Open Badges 3.0

EndorsementCredential



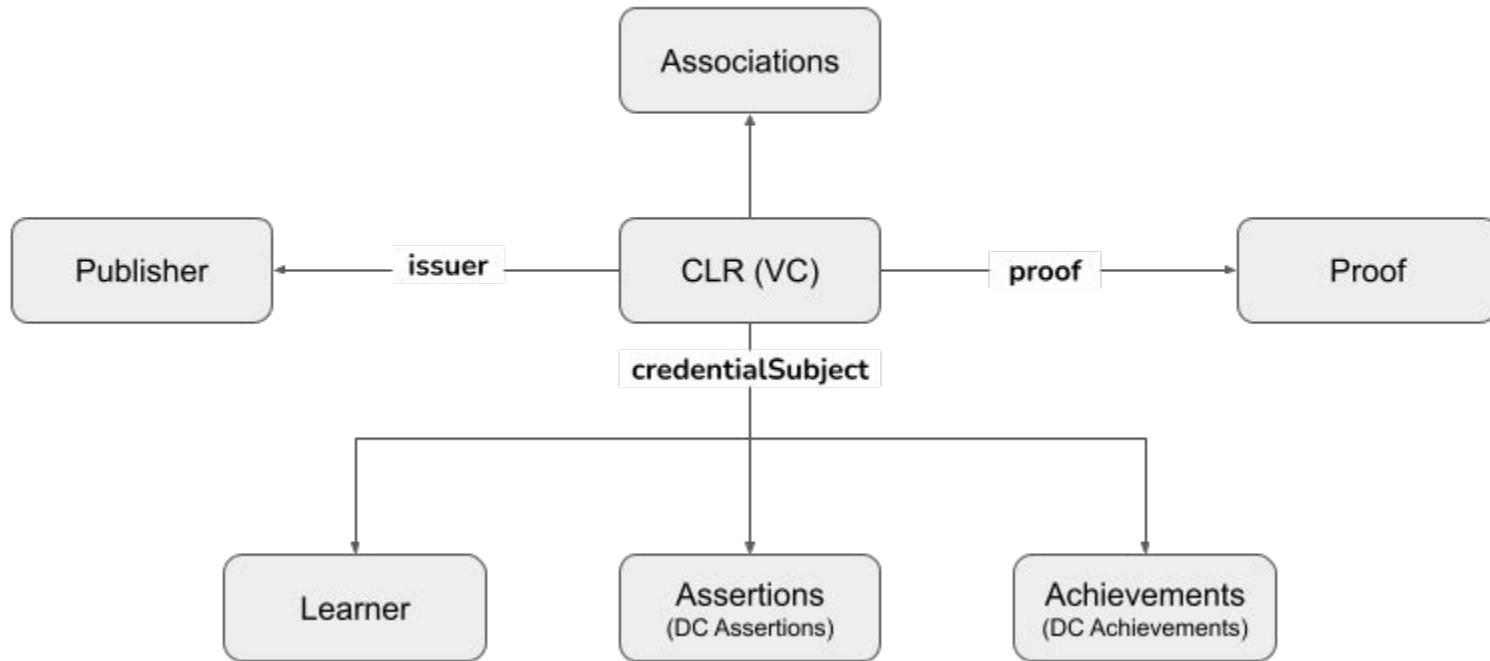
Open Badges 3.0

EndorsementSubject

- Type. MUST contain *EndorsementSubject*. (type)
- Writing about the claim (*endorsementComment*)

Comprehensive Learner Record 2.0

ClrCredential



Comprehensive Learner Record 2.0

ClrCredential

- Context. MUST contain, **in order**:
 - <https://www.w3.org/2018/credentials/v1>
 - <https://purl.imsglobal.org/spec/clr/v2p0/context.json>
 - <https://purl.imsglobal.org/spec/ob/v3p0/context-3.0.1.json>
- Type. MUST contain *ClrCredential* (type)
- Endorsements (JWT + embedded)
- Complete / partial (*partial*)

Comprehensive Learner Record 2.0

ClrSubject

- Type. MUST contain *ClrSubject* (type)
- Recipient. (id / identifier)
- Expected achievements. (*achievement*)
- Assertions / Awarded Credentials. (*verifiableCredential*)
- Semantic relationship between source and target achievements
(*association*)

Comprehensive Learner Record 2.0

Association

- Type. MUST be Association (*type*)
- Relationship type. (*associationType*)
- URI of the source. (*sourceId*)
- URI of the destination. (*targetId*)

Comprehensive Learner Record 2.0

AssociationType

- exactMatchOf
- isChildOf
- isParentOf
- isPartOf
- isPeerOf
- isRelatedTo
- precedes
- replacedBy



Proofs



OB 3.0 / CLR 2.0

Proofs

- JSON Web Token Proof - External (VC-JWT)
 - Linked Data Proofs - Embedded (*proof* property)
 - Public key dereferencing
-
- Multiple proofs MAY be provided

OB 3.0 / CLR 2.0

JSON Web Token Proof (VC-JWT)

- JSON Web Token [RFC7519]
- JSON Web Signature [RFC7515]
- Compact JWS string

header.payload.signature

OB 3.0 / CLR 2.0

JSON Web Token Proof (VC-JWT)

1. Create the JOSE Header, specifying the signing algorithm to use
2. Create the JWT Payload from the credential to be signed
3. Compute the signature of the JWT Payload
4. Encode the resulting JWS as a Compact JWS

OB 3.0 / CLR 2.0

JSON Web Token Proof (VC-JWT) - JOSE Header

- *alg* MUST be *RS256* as a minimum. Support for other algorithms are permitted.
- *kid* with the URI of the JWK representing the public key. (optional)
- *jwk* with a JWK representing the public key. (optional)
- *typ* MUST be *JWT* (optional)

OB 3.0 / CLR 2.0

JSON Web Token Proof (VC-JWT) - Payload

- *iss* with the *issuer.id* property of the Credential.
- *jti* with the *id* property of the Credential.
- *nbf* with the *issuanceDate* property of the Credential.
- *sub* with the *credentialSubject.id* property of the Credential.
- *vc* the the Credential.
- *exp* with the *expirationDate* property of the Credential. (optional)



CLR 2.0

JSON Web Token Proof (VC-JWT) - Payload with nested credentials

```
{  
  ...  
  "credentialSubject": {  
    "verifiableCredential": [  
      {  
        "id": "assertion1",  
        ...  
      },  
      {  
        "id": "assertion2",  
        ...  
      }  
    ]  
  }  
}
```



```
{  
  ...  
  "credentialSubject": {  
    "verifiableCredential": [  
      "header.payload.signature",  
      "header.payload.signature"  
    ]  
  }  
}
```



OB 3.0 / CLR 2.0

JSON Web Token Proof (VC-JWT) - Signature

1. Base64 Encode JOSE Header in UTF 8. (A)
2. Base64 Encode payload. (B)
3. Calculate signature of $A \parallel ":" \parallel B$ using alg and private key.
4. Base64 Encode of signature.

Base64(UTF8(JOSE Header)) $\parallel ":" \parallel$ Base64(payload) $\parallel ":" \parallel$ Base64(signature)

OB 3.0 / CLR 2.0

Linked Data Proofs

VC Data Integrity 1.0 (<https://www.w3.org/TR/vc-data-integrity>)

- Public / Private Key.
- Controller.
- Controller Document.
- Verification Method.
- Proof type.
- Proof purpose.



OB 3.0 / CLR 2.0

Linked Data Proofs - Embedded

- EdDSA Cryptosuite v2022.
 - <https://w3c.github.io/vc-di-eddsa/>
 - Key representation: Multibase / Multicodec
 - Key encoding: Multikey / Ed25519VerificationKey2020



OB 3.0 / CLR 2.0

Linked Data Proofs - Verification Method

- A way to dereference the public key.
- <https://www.w3.org/TR/vc-data-integrity/#retrieve-verification-method>
- Allowed several methods
 - HTTP
 - DID
 - ...



OB 3.0 / CLR 2.0

Linked Data Proofs - Verification Method - HTTP

```
{  
  '@context': 'https://w3id.org/security/multikey/v1',  
  'type': 'Multikey',  
  'controller': 'https://example.com/issuers/876543',  
  'id':  
    'https://example.com/issuers/876543#z6MkjZRZv3aez3r18pB1RBFJR1kwUVJ5jHt92JmQwXbd5hw1'  
,  
  'publicKeyMultibase': 'z6MkjZRZv3aez3r18pB1RBFJR1kwUVJ5jHt92JmQwXbd5hw1'  
}
```



OB 3.0 / CLR 2.0

Linked Data Proofs - Verification Method - HTTP - Controller Document

```
"@context": [  
    "https://www.w3.org/ns/did/v1",  
    "https://w3id.org/security/multiclient/v1"  
,  
  "id": 'https://example.com/issuers/876543',  
  "assertionMethod": [  
  
    'https://example.com/issuers/876543#z6MkjZRZv3aez3r18pB1RBFJR1kwUVJ5jHt92JmQwXbd5hwI'  
]
```

OB 3.0 / CLR 2.0

Linked Data Proofs - Creation

- Define the Multikey to use

```
const controller = 'https://example.com/issuers/876543'  
const keyPair = await Ed25519Multikey.from({  
  '@context': 'https://w3id.org/security/multikey/v1' ,  
  type: 'Multikey' ,  
  controller,  
  id: controller + '#z6MkjZRZv3aez3r18pB1RBFJR1kwUVJ5jHt92JmQwXbd5hw1' ,  
  publicKeyMultibase: 'z6MkjZRZv3aez3r18pB1RBFJR1kwUVJ5jHt92JmQwXbd5hw1' ,  
  secretKeyMultibase:  
  'zrv2bqTbNwCTsRrHFcJCPjVAduh4Ezcnoq1A3ZxH1GWTNkxipLvuaAoMFmze2gFN9oNXfJjufxSHWVZzsJiUsMHFMcx'  
});
```



OB 3.0 / CLR 2.0

Linked Data Proofs - Creation

- Create a DataIntegrityProof
 - <https://www.w3.org/TR/vc-data-integrity/#dataintegrityproof>
 - cryptosuite MUST be eddsa-2022
 - verificationMethod MUST be an URL to the verification method object

OB 3.0 / CLR 2.0

Linked Data Proofs - Creation

- Sign the credential with your key
 - <https://w3c-ccg.github.io/data-integrity-spec/#proof-algorithm>
 - *proofPurpose* MUST be *assertionMethod*.
 - Produces a *proof* object.
- Add it as the *proof* property of the credential

OB 3.0 / CLR 2.0

Linked Data Proofs - Creation



Example

<https://github.com/1EdTech/dc-bootcamp/tree/eddsa-2022>

Revocation



OB 3.0 / CLR 2.0

Verification

- 1EdTech Revocation List Status Method

- *credentialStatus* property
- type MUST be *1EdTechRevocationList*.
- *id* MUST be an URL.

```
{  
    "id": "http://example.edu/credentials/3732",  
    "type": ["Credential", "DegreeCredential"],  
    ...  
    "credentialStatus": {  
        "id": "https://example.edu/revocations/24",  
        "type": "1EdTechRevocationList"  
    }  
}
```



<http://www.imsglobal.org/spec/vcrl/v1p0/>

Refresh

OB 3.0 / CLR 2.0

Verification

- 1EdTech Verifiable Credential Refresh Service

- *refreshService* property
- type MUST be *1EdTechCredentialRefresh*.
- *id* MUST be an URL.

```
{  
    "id": "http://example.edu/credentials/3732",  
    "type": ["Credential", "DegreeCredential"],  
    "issuer": "https://example.edu/issuers/14",  
    ...  
    "refreshService": [  
        {"id": "https://example.edu/refresh/3732",  
         "type": "1EdTechCredentialRefresh"  
    ]  
}
```



<http://www.imsglobal.org/spec/vccr/v1p0/>



Portability

Baking an image

OB 3.0 / CLR 2.0

Baking (PNG)

- iTXt chunk with keyword **openbadgecredential**.
- Credential inside
 - JWT (External Proof)
 - JSON representation (Embedded Proof)

```
var chunk = new iTXt({
  keyword: 'openbadgecredential',
  compression: 0,
  compressionMethod: 0,
  languageTag: '',
  translatedKeyword: '',
  text: '{
    "@context": [
      "https://www.w3.org/2018/credentials/v1",
      "https://purl.imsglobal.org/spec/ob/v3p0/context-3.0.1.json"
    ],
    "id": "http://example.edu/credentials/3732",
    "type": ["VerifiableCredential",
    "OpenBadgeCredential"],
    "issuer": {
      "id": "https://example.edu/issuers/565049",
      "type": "IssuerProfile",
      "name": "Example University"
    },
    "issuanceDate": "2010-01-01T00:00:00Z",
    "credentialSubject": {
      "id": "did:example:ebfeb1f712ebc6f1c276e12ec21"
    },
    "proof": { }
  }
})
```



OB 3.0 / CLR 2.0

Baking (SVG)

- **xmlns:openbadges** attribute.
- **openbadges:credential** tag with the credential
 - **verify** attribute with JWT (External Proof)
 - JSON representation (Embedded Proof)

```
<?xml version="1.0" encoding="UTF-8"?>
<svg xmlns="http://www.w3.org/2000/svg"
  xmlns:openbadges="https://purl.imsglobal.org/ob/v3p0"
  viewBox="0 0 512 512">
<openbadges:credential>
  <![CDATA[
    {
      "@context": [
        "https://www.w3.org/2018/credentials/v1",
        "https://purl.imsglobal.org/spec/ob/v3p0/context-3.0.1.json"
      ],
      "id": "http://example.edu/credentials/3732",
      "type": ["VerifiableCredential", "OpenBadgeCredential"],
      "issuer": {
        "id": "https://example.edu/issuers/565049",
        "type": "IssuerProfile",
        "name": "Example University"
      },
      "issuanceDate": "2010-01-01T00:00:00Z",
      "credentialSubject": {
        "id": "did:example:ebfeb1f712ebc6f1c276e12ec21"
      },
      "proof": { }
    }
  ]]>
</openbadges:credential>
<!-- rest-of-image -->
</svg>
```



Verification

OB 3.0 / CLR 2.0

Verification

- The credential (or presentation) conforms to the specification.
- The proof method is satisfied.
- The status check succeeds.
- Verification of the recipient succeeds. (optional)
- Verification of endorsements succeeds.
- Verification of verifiable credentials succeeds. (CLR)

It doesn't imply evaluation of the truth of the claims



OB 3.0 / CLR 2.0

Verification

1. Extract the credential (if baked)
 - o PNG: iTxt chunk with keyword **openbadgcredential**
 - o SVG: **openbadges:credential** tag with the credential
 - verify attribute with JWT (External Proof)
 - JSON representation (Embedded Proof)
 - o Decode JWT, if needed

OB 3.0 / CLR 2.0

Verification

2. Check conformance to the specification

- Context.
- Types.
- Required attributes (JSON schema).
- *credentialSchema* of type *JsonSchemaValidator2019*.
- Identifier in *credentialSubject* (*id* or *identifier*)
- Accessible URL in profiles (*issuer*, *creator*, *parent* ...)

OB 3.0 / CLR 2.0

Verification

3. Check the proof method is satisfied (JWT).
 - a. Base64url-decode the JOSE Header.
 - b. Obtain the public key JWK
 - If the header includes a kid property, Dereference the kid value to retrieve the public key JWK.
 - If the header includes a jwk property, convert the jwk value into the public key JWK.
 - c. Use the public key JWK to verify the signature [RFC7515]
 - d. Base64url-decode the JWT Payload segment of the Compact JWS and parse it into a JSON object.
 - e. Convert the value of vc claim to an OpenBadgeCredential



OB 3.0 / CLR 2.0

Verification

3. Check the proof method is satisfied (JWT) (contd).
 - f. required *iss* claim MUST be the *issuer.id* of the credential.
 - g. required *sub* claim MUST be the *credentialSubject.id* of the credential.
 - h. required *nbf* claim as ISO8601 DateTime MUST be *issuanceDate* of the credential.
 - i. required *jti* claim MUST be the id of the credential.
 - j. optional *exp* claim as ISO8601 DateTime MUST be *expirationDate* of the credential.



OB 3.0 / CLR 2.0

Verification

3. Check the proof method is satisfied (Embedded).

- Proof verification algorithm VC Data Integrity 1.0
 - <https://w3c.github.io/vc-data-integrity/#verify-proof>
- verificationMethod defines how to get public key
 - HTTP
 - DID
- Cryptosuite (eddsa-2022) defines
 - Transformation
 - Hashing
 - Verification

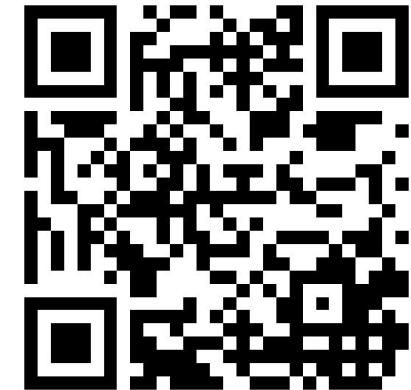


OB 3.0 / CLR 2.0

Verification

4. Refresh the credential

- *refreshService* of type *1EdTechCredentialRefresh*.
 - GET *id*
- Repeat steps 1, 2 & 3.
- Only refresh once.



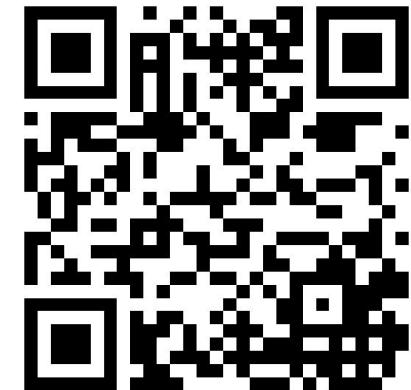
<http://www.imsglobal.org/spec/vccr/v1p0/>

OB 3.0 / CLR 2.0

Verification

5. Check status

- *credentialStatus* of type *1EdTechRevocationList*.
 - GET *id*
- Check *revokedCredentials* for a revoked credential with the credential's *id*
- Check the *revoked* boolean attribute of the revoked credential. Default to true
- Check the *revocationReason* attribute for the reason (optional)



<http://www.imsglobal.org/spec/vcr1/v1p0/>

OB 3.0 / CLR 2.0

Verification

5. Check status (cont)

- *issuanceDate*
- *expirationDate*

OB 3.0 / CLR 2.0

Verification

6. Verify recipient (optional)

- *credentialSubject.id* or *credentialSubject.identifier*
- Previously shared in a OOB process.
- Check *id*
- *identifier*
 - Known *identityType*
 - Can be hashed (*id + salt == identityHash*)
- Multiple identifiers provided

OB 3.0 / CLR 2.0

Verification

7. Verify endorsement credentials

- Extract the credential (if baked).
- Check conformance to the specification.
- Check the proof method is satisfied.
- Check status

CLR 2.0

Verification

8. Verify nested verifiable credentials

- Use OB 3.0 verification for *AchievementCredential*,
OpenBadgeCredential
- Do as appropriate for others.
 - <https://www.w3.org/TR/vc-data-model/#validation>



BadgeConnect API



OB 3.0 / CLR 2.0

API

- Exchange of credentials on behalf of holders
- RESTful
- OAuth 2.0
- Authorization Code Grant Flow
- Dynamic Registration
- API Base
 - /ims/ob/v3p0 (OB 3.0)
 - /ims/clr/v2p0 (CLR 2.0)



OB 3.0 / CLR 2.0

API

- {apiBase}/discovery
 - Service Description Document. Unprotected.
- {apiBase}/credentials
 - Get issued credentials from the resource server.
 - Create or replace a credential on the resource server.
- {apiBase}/profile
 - Fetch the profile from the resource server for the authenticated entity.
 - Update the profile for the authenticated entity.



OB 3.0

API - Scopes

- /ims/ob/v3p0/credentials
 - <https://purl.imsglobal.org/spec/ob/v3p0/scope/credential.readonly>
 - <https://purl.imsglobal.org/spec/ob/v3p0/scope/credential.upsert>
- /ims/ob/v3p0/profile
 - <https://purl.imsglobal.org/spec/ob/v3p0/scope/profile.readonly>
 - <https://purl.imsglobal.org/spec/ob/v3p0/scope/profile.update>

CLR 2.0

API - Scopes

- /ims/clr/v2p0/credentials
 - <https://purl.imsglobal.org/spec/clr/v2p0/scope/credential.readonly>
 - <https://purl.imsglobal.org/spec/clr/v2p0/scope/credential.upsert>
- /ims/clr/v2p0/profile
 - <https://purl.imsglobal.org/spec/clr/v2p0/scope/profile.readonly>
 - <https://purl.imsglobal.org/spec/clr/v2p0/scope/profile.update>

OB 3.0 / CLR 2.0

API - Service Discovery

- Open API definition file
- OAuth 2.0 Authorization Code Grant Flow details
 - *components/securitySchemes/OAuth2ACG/flows/authorizationCode*
- Custom fields:
 - *info/x-imssf-privacyPolicyUrl*. URL of the privacy policy.
 - *info/x-imssf-image*. URL of an image of the resource server.
 - *components/securitySchemes/OAuth2ACG/x-imssf-registrationUrl*. URL of the dynamic registration endpoint.
- Available scopes



OB 3.0 / CLR 2.0

API - Service Discovery

```
"info": {  
    "title": "OpenAPI schema for Open Badges",  
    "description": "Open Badges Data Model 3.0",  
    "termsOfService": "https://www.imsglobal.org/license.html",  
    "contact": {  
        "name": "1EdTech",  
        "url": "https://www.ledtech.org",  
        "email": "support@ledtech.org"  
    },  
    "license": {  
        "name": "1EdTech Specification Document License",  
        "url": "https://www.ledtech.org/license.html"  
    },  
    "version": "3.0",  
    "x-status": "Candidate Final",  
    "x-imssf-image": "https://www.ledtech.org/themes/custom/radix_sub/logo.svg",  
    "x-imssf-privacyPolicyUrl": "http://www.ledtech.org/privacystatement.html"  
}
```



OB 3.0 / CLR 2.0

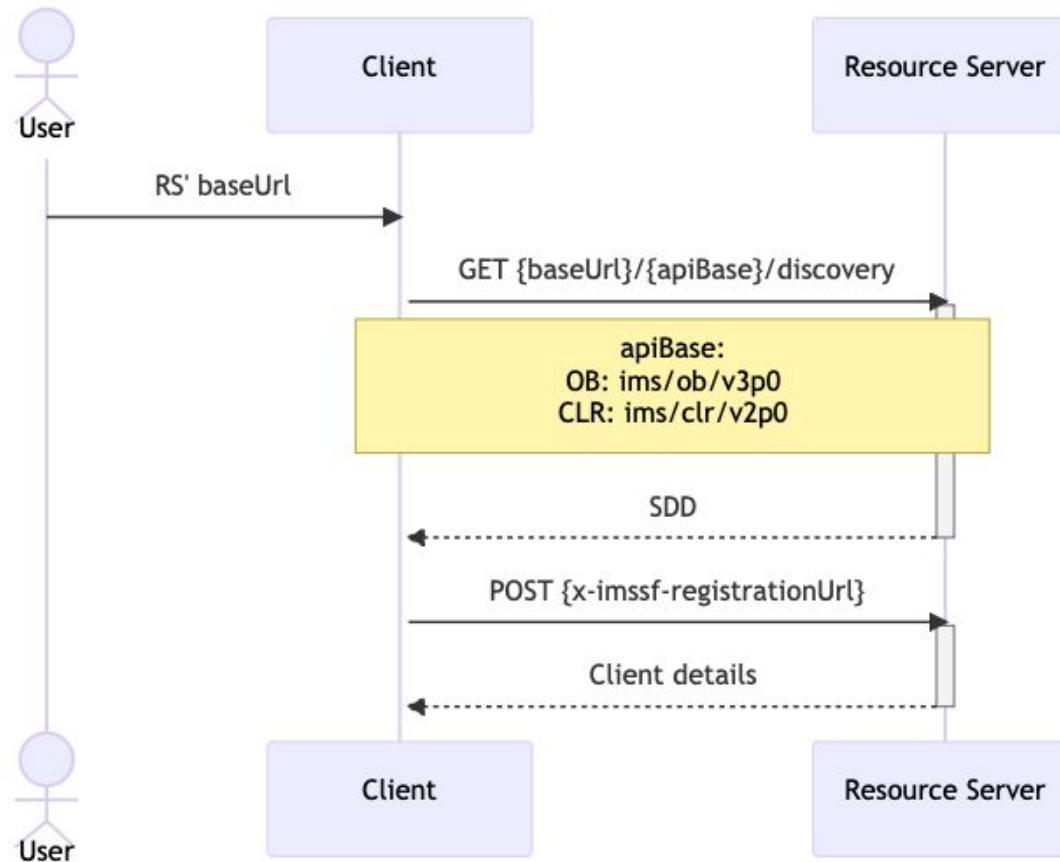
API - Service Discovery

```
"securitySchemes": {  
    "OAuth2ACG": {  
        "type": "oauth2",  
        "flows": {  
            "authorizationCode": {  
                "authorizationUrl": "https://1edtech.org/oauth/authorize",  
                "tokenUrl": "https://1edtech.org/oauth/token",  
                "refreshUrl": "https://1edtech.org//oauth/refresh",  
                "scopes": {  
                    "https://purl.imsglobal.org/spec/ob/v3p0/scope/credential.readonly": "...",  
                    "https://purl.imsglobal.org/spec/ob/v3p0/scope/credential.upsert": "...",  
                    "https://purl.imsglobal.org/spec/ob/v3p0/scope/profile.readonly": "...",  
                    "https://purl.imsglobal.org/spec/ob/v3p0/scope/profile.update": "..."  
                }  
            }  
        },  
        "x-imssf-registrationUrl": "https://1edtech.org/oauth/register"  
    }  
},  
{ver Learner Potential
```



OB 3.0 / CLR 2.0

API - Registration



OB 3.0 / CLR 2.0

API - Registration

Request

- client_name
- client_uri
- logo_uri
- tos_uri
- policy_uri
- software_id
- software_version
- redirect_uris
- scopes
- token_endpoint_auth_method (optional)
- grant_types (optional)
- response_types (optional)
- contacts (optional)

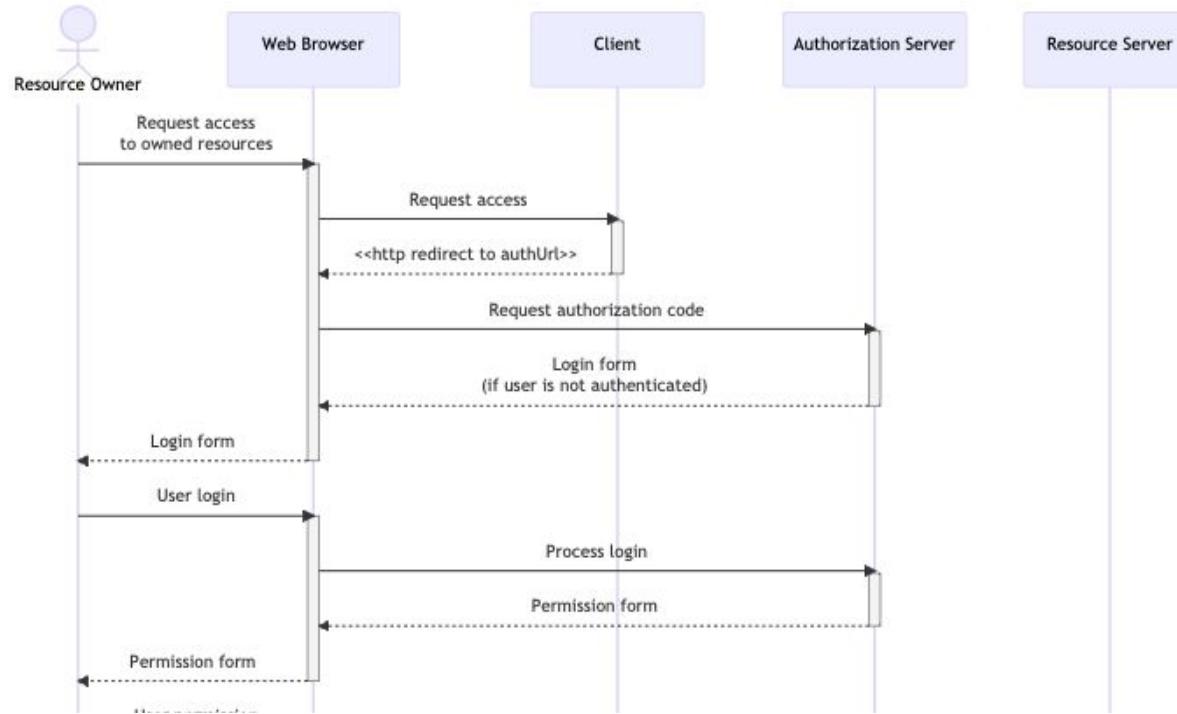
Response

- All provided information +
- client_id
- client_secret
- client_id_issued_at
- client_secret_expires_at



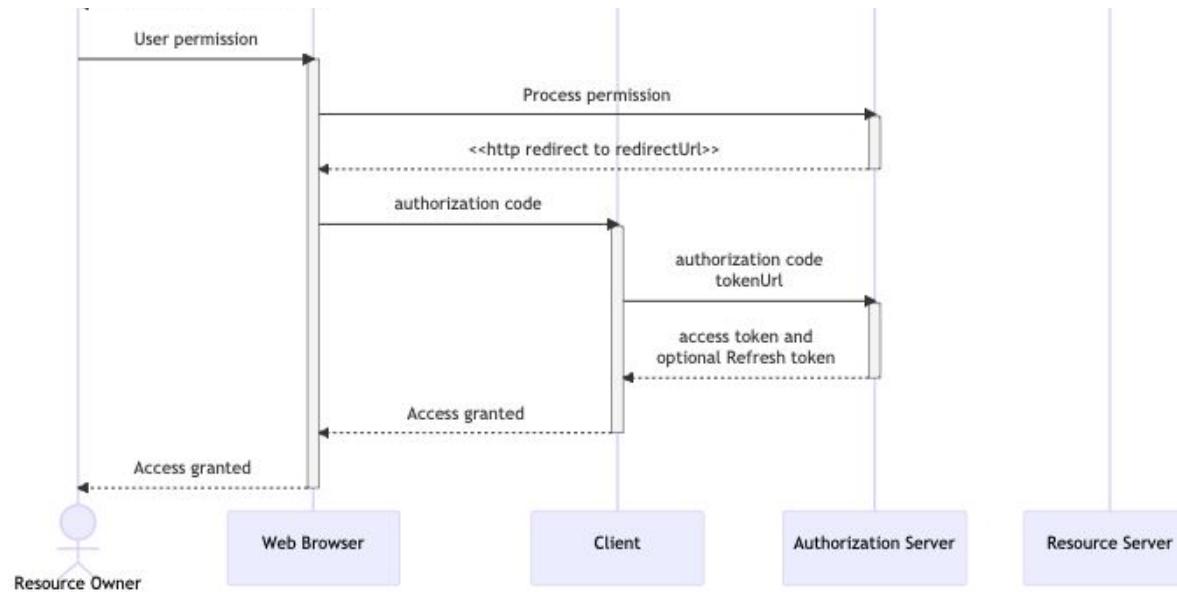
OB 3.0 / CLR 2.0

API - Obtaining Tokens



OB 3.0 / CLR 2.0

API - Obtaining Tokens



OB 3.0 / CLR 2.0

Credential equality and comparison

- Process to determine whether a verifiable credential is semantically equivalent to another one.
- A Host SHOULD treat a credential as the same as another when
 - *issuer.id* are equals*
 - *id* are equals*
- *issuanceDate* determines which is newer
 - *awardedDate* keeps the first issuance date

* after unescaping of any percent encoded characters [RFC3986] followed by truncation of leading and trailing whitespace

OB 3.0 & CLR 2.0

How to leverage the tight integration
between the two specifications



OB 3.0 / CLR 2.0

- Shared ancestor: W3C's Verifiable Credential
- CLR can hold verifiable credentials, including Open Badges 3.0's *AchievementCredential*
- Similar Proof creation.
- Shared API security model.
- Similar verification method.
- Similar endpoints.
- Similar comparison and equality algorithm.

W3C VC

How to leverage the alignment with the
W3C Verifiable Credentials standard



OB 2.0 / CLR 2.0

VC Extensions

- 1EdTech Revocation List Status Method
- 1EdTech Credential Refresh Service
- 1EdTech JSON Schema Validator 2019

<https://w3c.github.io/vc-specs-dir>



OB 2.0 / CLR 2.0

Connecting to Verifiable Credentials Wallets

- OB 3.0 and CLR 2.0 APIs serve use cases for web services to verifiably exchange credentials on behalf of holders.
- All Verifiable Credentials can and will be communicated over a range of protocols
 - CHAPI
 - DIDCom
 - OpenID4VC

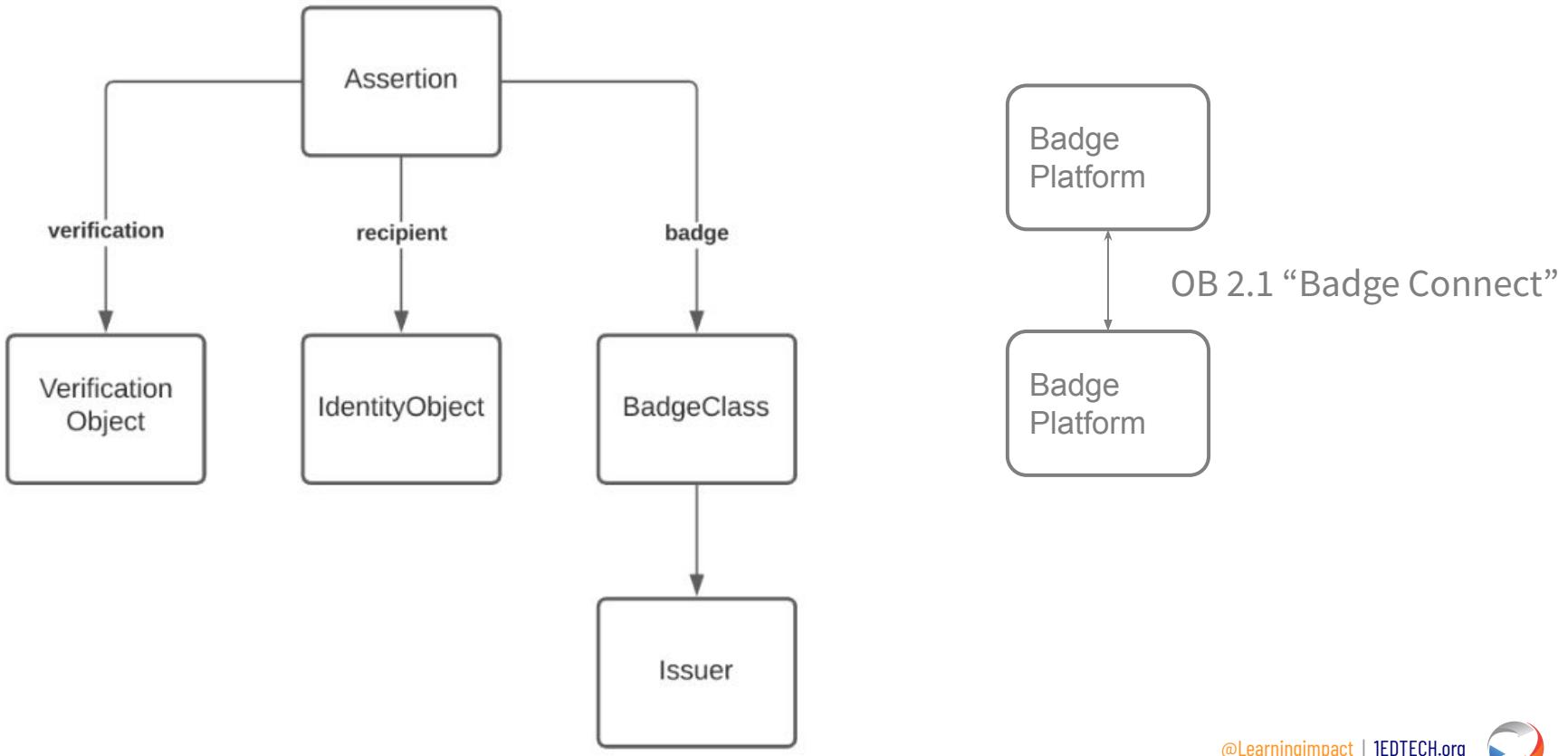


Upgrade

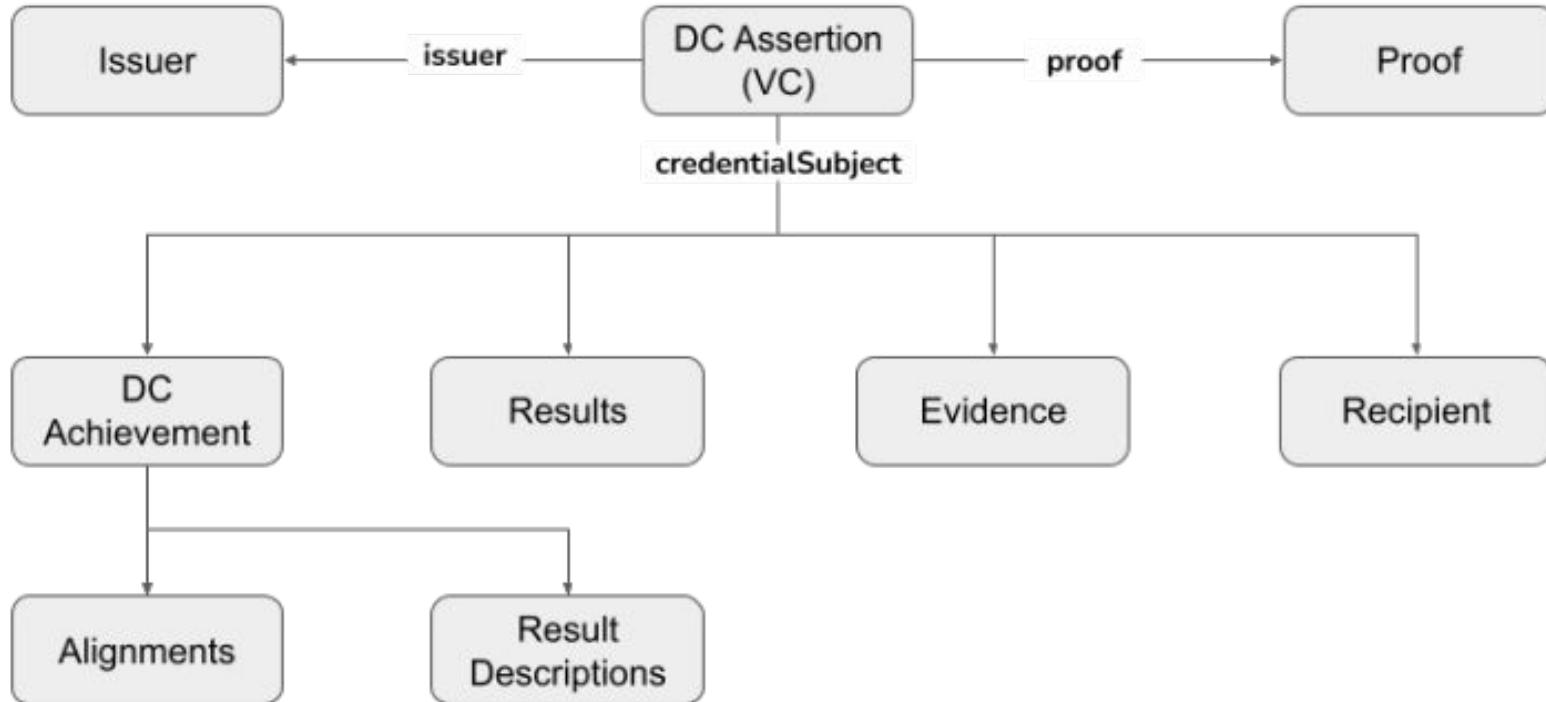
How to migrate from OB 2.0/2.1 and CLR
1.0



OpenBadges 2.0 and 2.1



OpenBadges 3.0



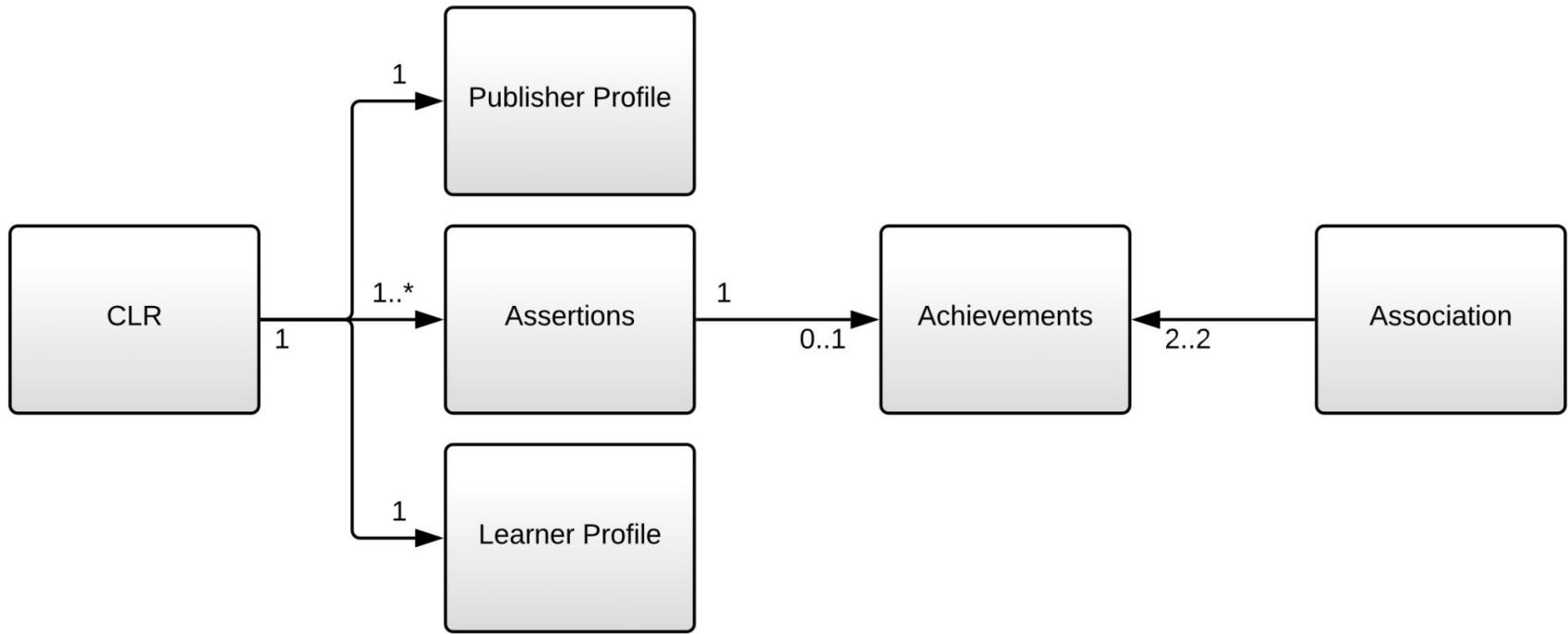
OB 2.0 / OB 3.0

Upgrading

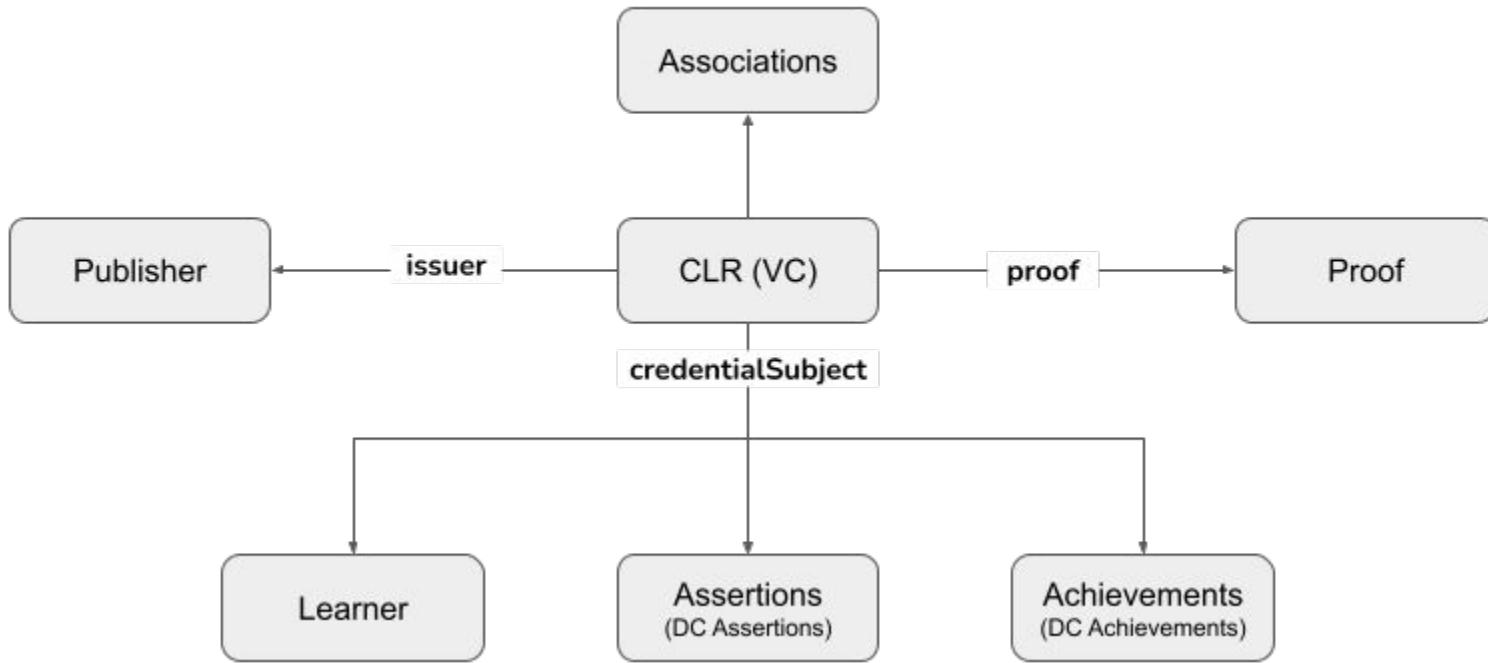
- Assertion
- BadgeClass
- VerificationObject
- Endorsement
- Profile
 - RevocationList
 - publicKey
 - verification
- OpenBadgeCredential
- Achievement
- Proof
- EndorsementCredential
- Profile
- CredentialStatus
- Proof
- Proof



Comprehensive Learner Record 1.0



Comprehensive Learner Record 2.0



CLR 1.0 / CLR 2.0

Upgrading

- CLR
- Assertion
- Achievement
- Association
- Verification
- Endorsement
- Profile
 - RevocationList
 - publicKey
 - verification
- ClrCredential
- VerifiableCredentials
- Achievement
- Association
- Proof
- EndorsementCredential
- Profile
 - CredentialStatus
 - Proof
 - Proof



Versions Coexistence

Open Badges

OB 2.0 & OB 3.0

- Tons of OB 2.0 issued
- Same achievement data may be exposed in OB 2.0 and OB 3.0/CLR 2.0 formats
- Not advisable to publish a combined OB 2.0 & 3.0 entity.
- Recommended to link OB 2.0 and OB 3.0 versions.
 - *related* property in Achievement.

<https://www.imsglobal.org/spec/ob/v3p0/impl/#how-to-support-both-ob-2-0-and-ob-3-0-as-an-issuer>



OB 2.0 & OB 3.0

OB 2.0 in OB 3.0

```
{  
    "@context": "https://purl.imsglobal.org/spec/ob/v3p0/context-3.0.1.json" ,  
    "type": [ "Achievement" , "https://w3id.org/openbadges#BadgeClass" ] ,  
    "@language": "en" ,  
    "id": "https://example.com/achievements/c3c1ea5b-9d6b-416d-ab7f-76da1df3e8d6"  
    ...  
    "related": [  
        {  
            "type": [  
                "Related" ,  
                "https://w3id.org/openbadges#BadgeClass"  
            ] ,  
            "id": "https://example.com/badgeclasses/c3c1ea5b-9d6b-416d-ab7f-76da1df3e8d6" ,  
            "version": "Open Badges v2p0"  
        } ,  
        {  
            "type": [ "Related" ] ,  
            "id": "https://example.com/achievements/c3c1ea5b-9d6b-416d-ab7f-76da1df3e8d6/es" ,  
            "@language": "es"  
        }  
    ]  
}
```



OB 2.0 & OB 3.0

Issuer: OB 2.0 in OB 3.0

```
{  
  "@context": [  
    "https://www.w3.org/ns/did/v1" ,  
    "https://www.w3.org/2018/credentials/v1" ,  
    "https://purl.imsglobal.org/spec/ob/v3p0/context-3.0.1.json" ,  
    "https://w3id.org/security/suites/ed25519-2020/v1"  
  ],  
  "id": "did:web:example.com:issuers:540e388e-2735-4c3e-9709-80142801c774" ,  
  "alsoKnownAs": "https://example.com/issuers/v2p0/540e388e-2735-4c3e-9709-80142801c774" ,  
  "otherIdentifier": [{  
    "type": ["IdentifierEntry"] ,  
    "identifier": "https://example.com/issuers/v2p0/540e388e-2735-4c3e-9709-80142801c774" ,  
    "identifierType": "identifier"  
  }]  
  ...  
}
```



OB 2.0 & OB 3.0

OB 3.0 in OB 2.0

```
{  
  "@context": "https://w3id.org/openbadges/v2" ,  
  "type": "BadgeClass" ,  
  "id": "https://example.com/badgeclasses/c3c1ea5b-9d6b-416d-ab7f-76da1df3e8d6" ,  
  "related": [ {  
    "type": [ "https://purl.imsglobal.org/spec/vc/ob/vocab.html#Achievement" ] ,  
    "id": "https://example.com/achievements/c3c1ea5b-9d6b-416d-ab7f-76da1df3e8d6" ,  
    "version": "Open Badges v3p0"  
  } ]  
}
```



OB 2.0 & OB 3.0

Issuer: OB 3.0 in OB 2.0

```
{  
  "@context": "https://w3id.org/openbadges/v2" ,  
  "id": "hhttps://example.com/issuers/v2p0/540e388e-2735-4c3e-9709-80142801c774" ,  
  "type": "Profile" ,  
  "name": "Example Institution" ,  
  "url": "https://example.com" ,  
  "email": "info@example.com" ,  
  "related": [ {  
    "type": [  
      "https://purl.imsglobal.org/spec/vc/ob/vocab.html#Profile"  
    ] ,  
    "id": "did:web:example.com:issuers:540e388e-2735-4c3e-9709-80142801c774" ,  
    "version": "Open Badges v3p0"  
  } ]  
}
```



Versions Coexistence

Comprehensive Learner Record



CLR 2.0

Versions Coexistence

- More expensive than OB
- Less ecosystem of consumption
- Not likely to be worth the investment
- May use OB version coexistence for individual achievements.

Certification

Conformance and certification
expectations and how to use the
Conformance Test Systems



Certification Guides



Open Badges 3.0

[http://www.imsglobal.org/spec/ob/v3p0/cert](https://www.imsglobal.org/spec/ob/v3p0/cert)

Power Learner Potential



Comprehensive Learner Record 2.0

<https://www.imsglobal.org/spec/clr/v2p0/cert>

@Learningimpact | IEDTECH.org

© 2022 1EdTech Consortium, Inc. All rights reserved.



Certification

Profiles

- Issuer
 - Issuer Only
 - Service Consumer Write
- Displayer
- Host
 - Service Consumer Read
 - Service Provider Read
 - Service Provider Write

Consumer (Initiates Requests)	Provider (Responds to Requests)
<u>Service Consumer (Read)</u>	<u>Service Provider (Read)</u>
<u>Service Consumer (Write)</u>	<u>Service Provider (Write)</u>



Certification Suite

<https://certification.imsglobal.org/certification/verifiable-credentials>

- Members Only
 - Contributing
 - Affiliates
 - Digital Credentials and Badges Alliance
- OB 3.0 & CLR 2.0





Welcome to 1EdTech testing system

Open Badges 3.0 Certification

The goal of IMS certification for Open Badges is to ensure interoperable implementations of badging systems that generate and issue digital badges as well as those that host and display badges.

[⊕ Certify](#)

Comprehensive Learner Record 2.0 Certification

The goal of certification for Comprehensive Learner Record (CLR) is to ensure interoperable implementations of credentialing systems that generate and issue ClrCredentials as well as those that host and display ClrCredentials.

[⊕ Certify](#)





Open Badges 3.0 Certification - Open Badges Issuer certification

Issue a valid
Open Badge

Register your client to
use the API

Endpoint and OAuth registration for
consumer calls

Endpoint for sending a
credential

Endpoint for
sending a profile

Issue a valid Open Badge

Create a valid baked 3.0 badge and issue it to the recipient conformance@imsglobal.org and submit the issued badge to the conformance test system.

More Info

Badge

The uploaded badge must be valid and verifiable

Uri to Resource

Video

Demonstrate through video the candidate platform's methodology for a recipient to retrieve their badge

Uri to Resource

Submit





Open Badges 3.0 Certification - Open Badges Displayer certification

Display of a badge

The candidate platform must verify these badges and submit the status in the conformance test system. Among these badges there should be a valid one, an expired one and a revoked one.

More Info

- <https://certification.imsglobal.org/certification/verifiable-credentials/vc/credentials/ob-1>
- <https://certification.imsglobal.org/certification/verifiable-credentials/vc/credentials/ob-2>
- <https://certification.imsglobal.org/certification/verifiable-credentials/vc/credentials/ob-3>

Video of first badge

Demonstrate through separate video that the platform allows viewers of badges to see the following data in provided badge.

Video of second badge

Demonstrate through separate video that the platform allows viewers of badges to see the following data in provided badge.

Video of third badge

Demonstrate through separate video that the platform allows viewers of badges to see the following data in provided badge.



certification.imsglobal.org

https://certification.imsglobal.org/certification/verifiable-credentials/run

https://certification.imsglobal.org/certification/verifiable-credentials/vc/credentials/ob-1

1EDTECH

Open Badges 3.0 Certification - Host Service Conformance

Register your client to use the API	Endpoint and OAuth registration for consumer calls	Consumer call to getCredentials	Consumer call to getProfile with filter and query	Consumer call to getProfile	Provider Setup	OAuth client registration	OAuth authorization flow for provider calls	Provider call to getCredentials	Provider call to getCredentials	Provider call to getProfile	Provider call to upsertCredentials	Provider call to putProfile
-------------------------------------	--	---------------------------------	---	-----------------------------	----------------	---------------------------	---	---------------------------------	---------------------------------	-----------------------------	------------------------------------	-----------------------------

Register your client to use the API

Exchange of endpoint information and client registration

API Base URI: <https://certification.imsglobal.org/certification/verifiable-credentials/sessions/b00d8825-249f-4d7d-a1b1-565c49e43bee/ims/ob/v3p0>

REST Call to /sessions/b00d8825-249f-4d7d-a1b1-565c49e43bee/oauth/register

Please make a **POST** call to the endpoint **/sessions/b00d8825-249f-4d7d-a1b1-565c49e43bee/oauth/register**. The test expects a **200** response code.

The application will update the test result automatically

Messages

22/3/2023 17:37:19 Connected. Listening to events for current session (b00d8825-249f-4d7d-a1b1-565c49e43bee)

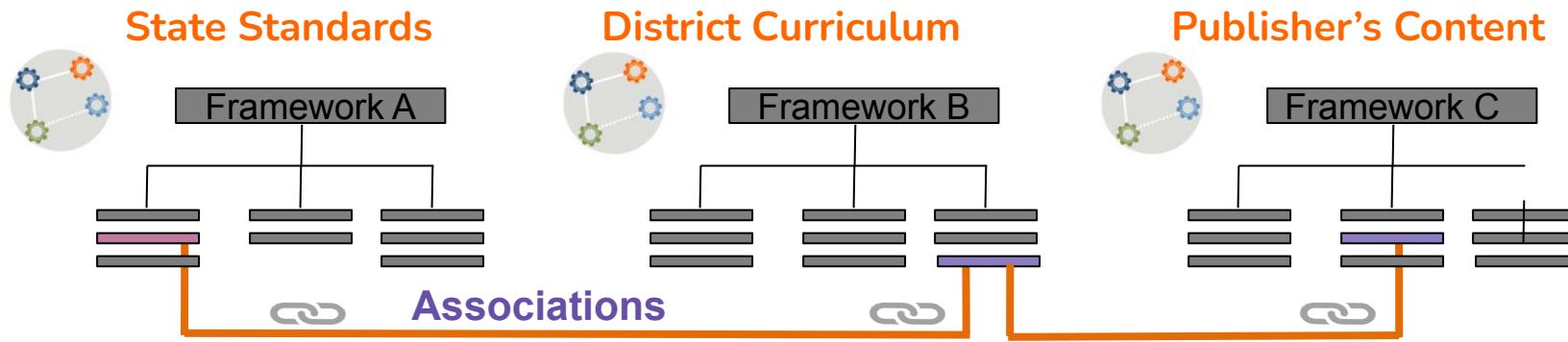


DC in 1EdTech Ecosystem

The relationship between UB and ULR
and the broader 1EdTech Ecosystem
including how to integrate with IMS
CASE.



CASE Connects Academic Standards

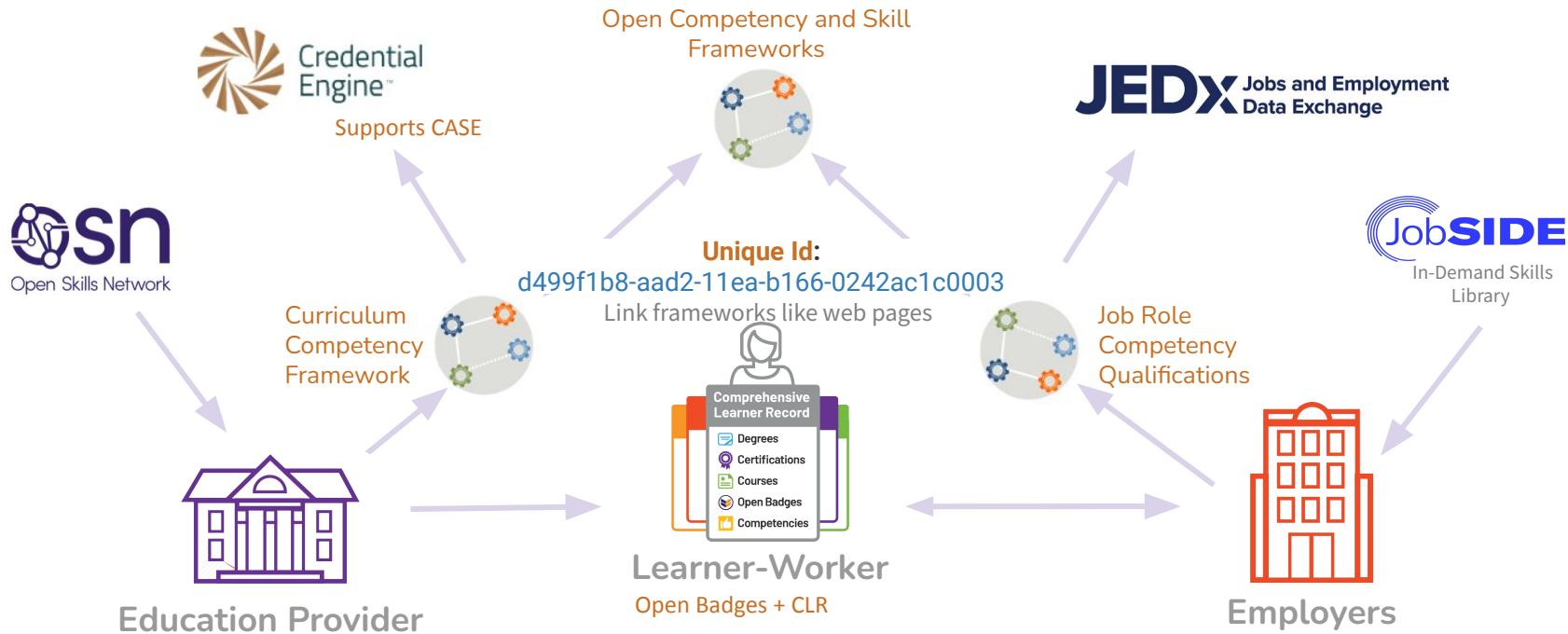


States publish Academic Standards that School Districts align their programs to, and publishers align their content and assessments



CASE Connects an Expanding Credentials Ecosystem

An open standard for publishing frameworks in a digital form for interconnection



OB 3.0 / CLR 2.0

Achievement Alignment

```
...
  "achievement": {
    "id": "https://1edtech.edu/achievements/1" ,
    "type": "Achievement",
    "criteria": {
      "narrative": "Cite strong and thorough textual evidence to support analysis of what the text says explicitly as well as inferences drawn from the text, including determining where the text leaves matters uncertain"
    },
    "description": "Analyze a sample text" ,
    "name": "Text analysis" ,
    "alignment": [
      {
        "type": "Alignment",
        "targetCode": "ce-cf4dee18-7cea-443a-b920-158a0762c6bf" ,
        "targetFramework": "Edmonds College Course Catalog" ,
        "targetName": "Requirements Analysis" ,
        "targetType": "ceterms:Credential" ,
        "targetUrl": "https://credentialfinder.org/credential/20229/Requirements_Analysis"
      }
    ]
  }
...

```



THANK YOU

For more information:

Xavi Aracil Diaz
Technical Standards Architect
xaracil@1edtech.org

