

INTERNET2 Attribute Registry Overview

The Attribute Registry V 1.0 is one of the early deliverables from the Scalable Privacy project.

The core data elements are attributes, each of which comes from one of a defined set of specifications or standards. The images in this overview were taken directly from the ontology tool, Protégé, used to maintain the registry.

- Specifications
- Attributes
- Attribute Class
- Future Work
- Reference: The Currently Defined Set of Forty-six Attribute Classes

Specifications

The screenshot shows the Protégé ontology editor interface. On the left, the 'Class hierarchy: Specification' panel shows a tree structure with 'Thing' as the root, followed by 'Attribute', 'AttributeClass', and 'Specification' (highlighted in blue). The main area displays the 'Description: Specification' panel, which includes sections for 'Annotations', 'Equivalent To', 'SubClass Of', and 'Members'. The 'Members' section lists various attribute classes associated with the 'Specification' class, each with a diamond icon and a set of control icons (question mark, at-sign, and X).

Member Class	Control Icons
eduCourse	? @ X
eduMember	? @ X
eduPerson	? @ X
FICAM	? @ X
IMS_Person	? @ X
OpenID_Connect	? @ X
OpenSocial	? @ X
RFC4524	? @ X
Schac	? @ X
SCIM_Core	? @ X
VA_EDM	? @ X
X.520	? @ X

Each attribute is associated with the specification that defines it. For example, the OpenID Connect specification covers the following attributes:

The screenshot displays the Attribute Registry interface for the **OpenID_Connect** class. On the left, the **Class hierarchy** shows **Thing** as the parent class, with **Attribute**, **AttributeClass**, and **Specification** as subclasses. Below this, the **Object property hierarchy** lists **topObjectProperty** with sub-properties: **classifies**, **isClassifiedBy**, **isSpecifiedBy**, and **specifies**.

The main content area is divided into three panels:

- Annotations: OpenID_Connect**: Shows a single annotation with a plus sign to add more.
- Description: OpenID_Conn**: Shows the type **Specification** and options for **Same Individual As** and **Different Individuals**.
- Property assertions: OpenID_Connect**: Lists 25 object property assertions, all of type **specifies**, including: **oidc-website**, **oidc-preferred_username**, **oidc-region**, **oidc-phone_number**, **oidc-formatted**, **oidc-middle_name**, **oidc-profile**, **oidc-nickname**, **oidc-gender**, **oidc-locality**, **oidc-family_name**, **oidc-zoneinfo**, **oidc-country**, **oidc-locale**, **oidc-address**, **oidc-name**, **oidc-picture**, **oidc-email_verified**, **oidc-user_id**, **oidc-street_address**, **oidc-email**, **oidc-given_name**, **oidc-updated_time**, **oidc-postal_code**, and **oidc-birthday**.

Attributes

An example attribute entry in the registry appears as follows (this is the Profile attribute from OpenID Connect):

The screenshot shows the metadata for the **oidc-profile** attribute. The **Description** panel shows the type **Attribute**. The **Property assertions** panel shows two object property assertions: **isSpecifiedBy** (pointing to **OpenID_Connect**) and **isClassifiedBy** (pointing to **Profile**). The **Data property assertions** panel lists several metadata elements:

- AttributeName**: "profile"^^string
- SourceLocation**: "http://openid.net/specs/openid-connect-basic-1_0.html#id_res"^^string
- Syntax**: "string"^^string
- Section**: "2.52"^^string
- Definition**: "URL of the End-User's profile page."^^string

Note the metadata (Object properties and Data properties) recorded in the registry for the Profile attribute. Version 1.0 contains a minimal set of metadata elements. Other types of metadata may be added to suit emerging needs in the attribute ecosystem work.

Another example attribute entry in the registry is eduPersonPrincipalName from the eduPerson specification:

Property assertions: eduPersonPrincipalName

Object property assertions +

- isClassifiedBy Identifier
- isSpecifiedBy eduPerson

Data property assertions +

- SourceLocation "http://macedir.org/specs/eduperson/#eduPersonPrincipalName"^^string
- AttributeName "eduPersonPrincipalName"^^string
- Oid "1.3.6.1.4.1.5923.1.1.1.6"^^string
- Definition "scoped identifier for a person. It should be represented in the form 'user@scope' where 'user' is a name-based identifier for the person and where 'scope' defines a local security domain. Each value of 'scope' defines a namespace within which the assigned identifiers **MUST** be unique. Given this rule, if two eduPersonPrincipalName (ePPN) values are the same at a given point in time, they refer to the same person"^^string
- Syntax "directoryString"^^string
- Section "2.2.8"^^string

Note the relatively full Definition element in this case. This is drawn from the specification itself.

Attribute Class

At the top of the eduPersonPrincipalName example above, there is the object property "isClassifiedBy" with the value "Identifier". This is an example of a metadata element meant to categorize attributes across specifications into a defined set of types. This metadata element is called "Attribute Class". Here is the first part of a listing of the currently defined attribute classes:

Class hierarchy: Attr

- Thing
 - Attribute
 - AttributeClass**
 - Specification

Annotations: AttributeClass

Annotations +

Description: AttributeClass

Members +

- Account
- Address
- Affiliation
- Application
- Assurance
- Birthdate
- BirthPlace
- Citizenship
- Contact
- Country
- Date

A couple examples will clarify the notion of attribute class. Take the example of attributes relating to preferences.

The Open Social specification (attributes whose prefix is "osoc" contains two attributes in the class "preference". The LDAP specification (currently RFC4524) contains an attribute "drink" which indicates personal preference as well. Here is the preference attribute, osoc-emails-primary:

Another example of the attribute class metadata is "role". Several specifications (SCIM, SCHAC, LDAP and X.520) contain attributes meant to carry some definition of a person's role:

Future Work

Going forward, attributes from additional specifications and standards will be added (including a master set from FICAM and one from the state of Virginia). One open issue is whether the current list of attribute metadata is adequate or whether there would be value in carrying additional metadata elements in a general purpose registry of this sort.

Reference: The Currently Defined Set of Forty-six Attribute Classes

Account, Address, Affiliation, Application, Assurance
 Birthdate, BirthPlace
 Citizenship, Contact, Country
 DN, Date, Description
 Email, EmailMetadata, Entitlement
 Format
 Gender, Group
 HostName
 Identifier
 Language, Link, Locale, Location
 Manager
 Name
 Organization
 Password, Phone, Photo, Pointer, Position, Preference, Presence, Privacy, Profile
 Role
 Salutation, Search, StateOrProvince, StatusMessage, SuperiorNode
 Tag, Timezone
 URL

[Add Comment](#)