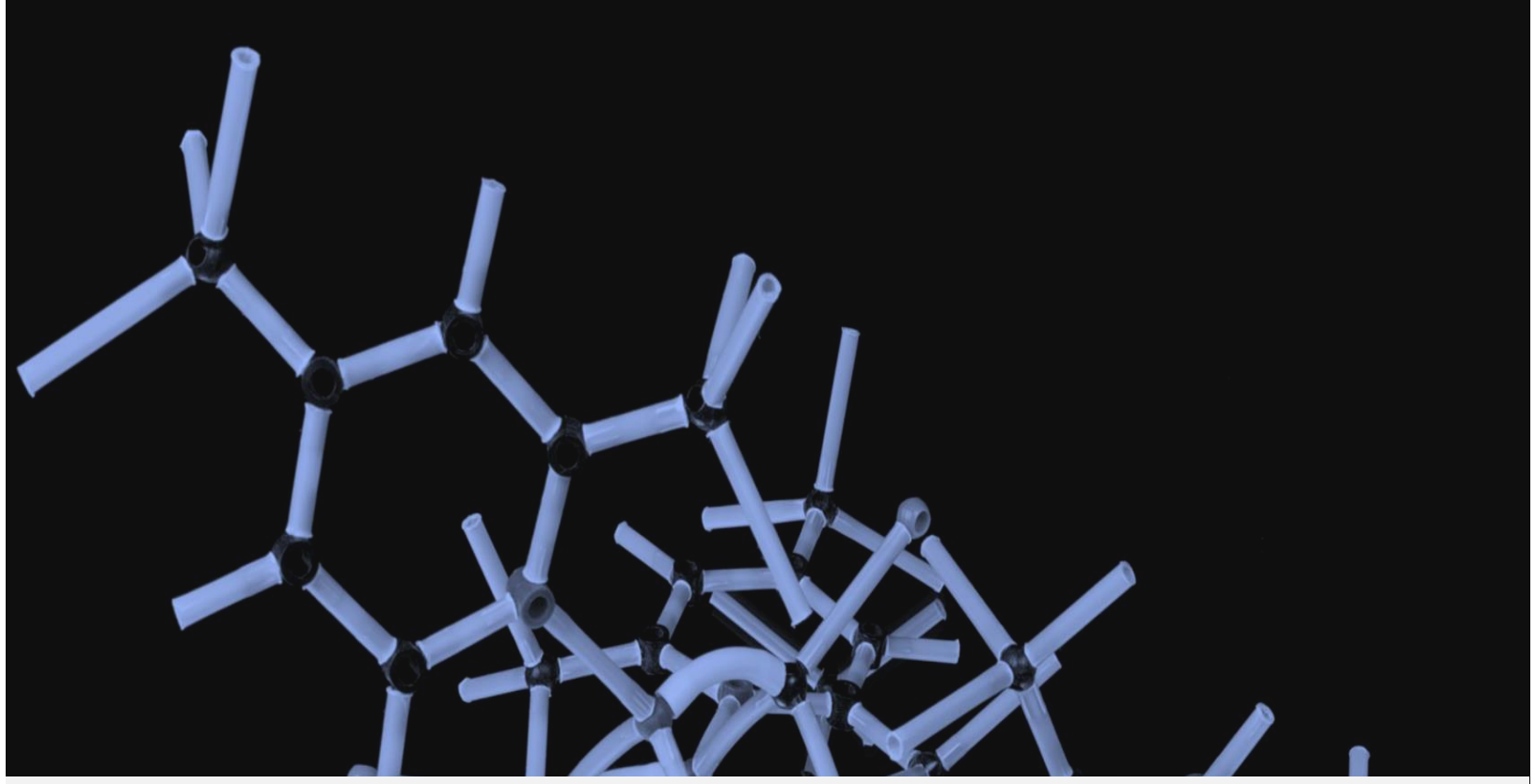
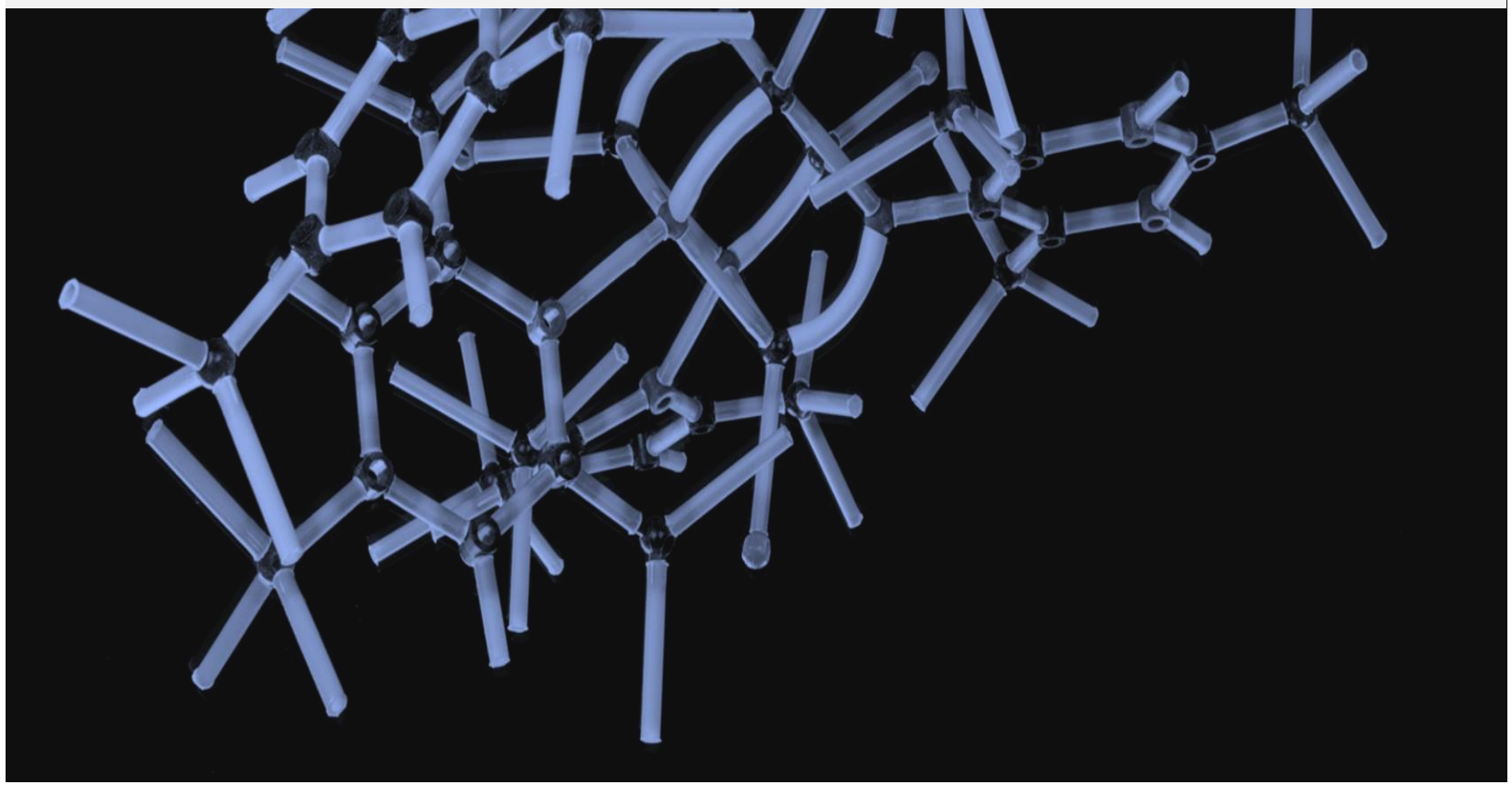


OSC ONTOLOGY

CEDS Ontology Guide

Version 12.0.0.0

May 2024



Introduction

The CEDS Ontology is being released as a draft for community review. The impetus for the creation of the Ontology is the result of multiple State Education Agency use cases that require the use of CEDS in a machine-readable language, such as JSON, JSON-LD, and XML. A workgroup was formed in May of 2022 to begin discussions related to the creation of JSON or XML. During that workgroup the decision was made to start the project by creating an ontology for CEDS which is the most granular description available. From that ontology, CEDS can then be expressed as JSON, JSON-LD, XML, and several other recognized standards.

The purpose of this document is to help explain some key concepts related to the Ontology during its initial review period. There is an expectation that many of the concepts, naming conventions, and diagrams in this document will change based off feedback from the CEDS community. As such, implementation of the Ontology during this draft review period is not recommended except for purposes of testing and providing feedback to CEDS.

The CEDS Ontology is comprised of six types of entities:

1. **Classes** - A Class represents a category, group, or set of individuals (also called instances) that share common characteristics, attributes, or behaviors. It serves as a conceptual grouping that defines the common features that its instances share. Classes are fundamental building blocks for creating ontologies, allowing you to organize and structure information in a logical and hierarchical manner within the semantic web context.
2. **Object Properties** - An Object Property represents a binary relation that exists between individuals (also known as instances) of different classes. Object properties are used to describe connections or links between instances, indicating some form of relationship, association, or connection between them. These properties enable the modeling of more complex relationships and interactions within ontologies, contributing to the rich representation of knowledge and data in the semantic web.
3. **Data Properties** - A Data Property represents a binary relation that associates individuals (instances) with literal values or data values. Unlike object properties, which link individuals to other individuals, data properties connect individuals to data values such as strings, numbers, dates, and other literals. Data properties are used to attribute specific data values to instances, allowing for the representation of attributes, characteristics, or data-related information within ontologies on the semantic web.
4. **Annotation Properties** - An Annotation Property is used to associate metadata or additional information with various elements within an ontology. Unlike object or data properties that relate individuals and data values, annotation properties are used to attach descriptive notes, comments, or labels to classes, individuals, properties, and other components of the ontology. These annotations do not affect the logical reasoning of the ontology but provide contextual information that aids in understanding, documentation, and communication of the ontology's content and structure.
5. **Datatypes** - A Datatype refers to a specific data type or format that specifies the kind of values that a data property can have. Datatypes define the permissible range of literal values that can be associated with individuals through data properties. These values can include strings, numbers, dates, Boolean values, and other well-defined data formats. Datatypes are crucial for

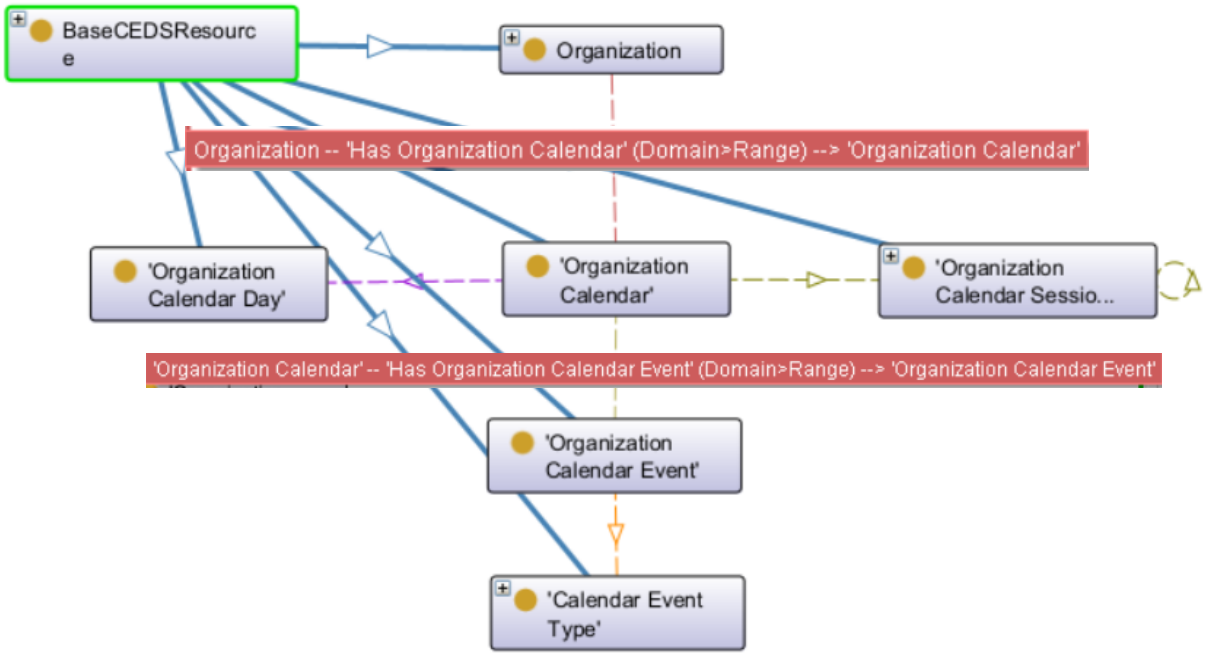
accurately representing and specifying the type of data that can be used in conjunction with data properties in an ontology.

6. **Individuals** - An Individual refers to a specific instance or entity that belongs to a particular class within an ontology. Individuals are concrete members of a class, representing distinct entities with their own identity and properties. They can be thought of as the actual instances or objects that the ontology describes, and they are used to represent real-world entities or concepts within the domain being modeled.

Examples

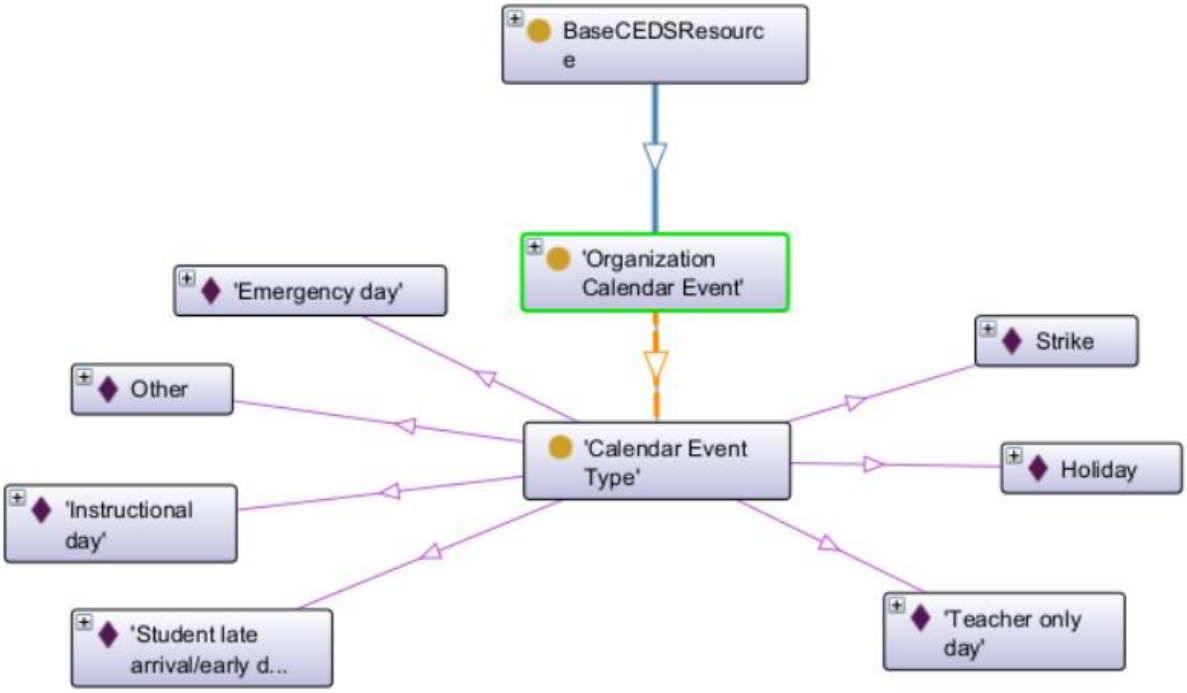
Example 1: Classes & Object Properties: Organization, Organization Calendar, and Organization Calendar Event

In the example below, we have multiple classes that relate to each other. Specifically, the class called "Organization" relates to the class called "Organization Calendar" through the object property "Has Organization Calendar." The class "Organization Calendar" relates to "Organization Calendar Event" through the object property "Has Organization Calendar Event."



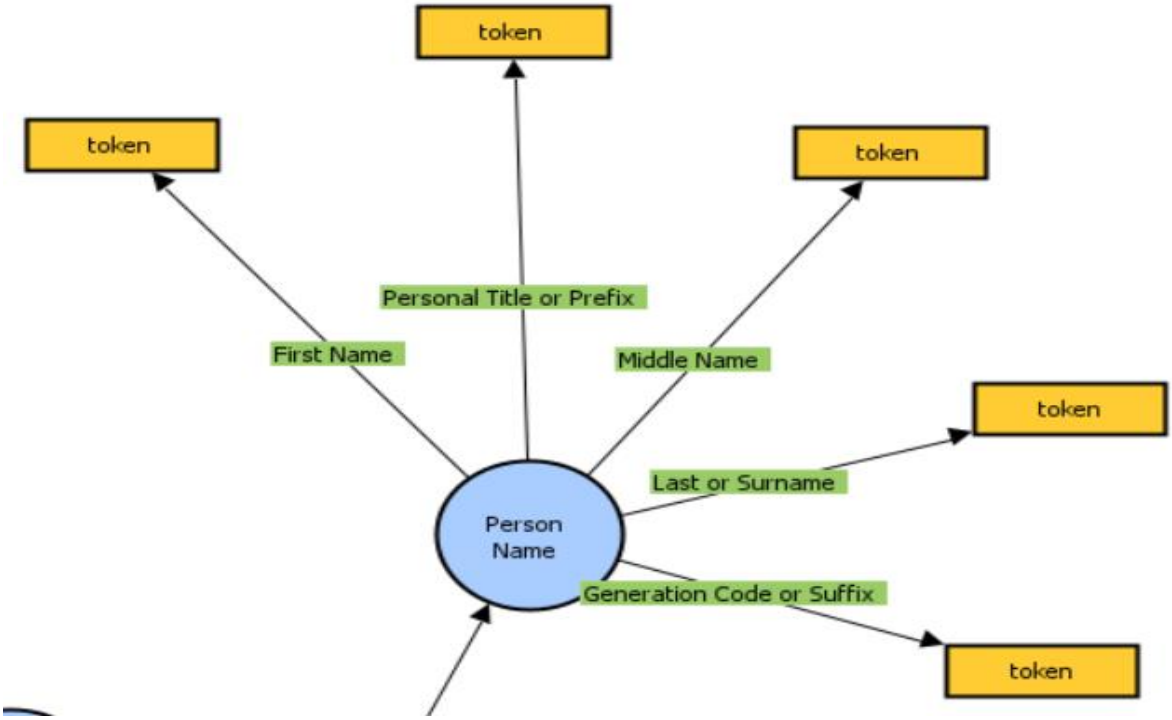
Example 2: Class and Individuals

In the previous example we showed how Organization Calendar Event relates back an Organization. In this example, we see that Organization Calendar Event relates to the class Calendar Event Type which relates directly to Individuals (standard option sets defined in CEDS). All CEDS Option Sets are set up as Individuals within the Ontology that relate directly to a Class named after the CEDS element that contains those options.



Example 3: Data Properties

In this example, see the Class called Person Name with Data Properties extending from it. Data Properties represent elements within CEDS that do not have standard option sets but rather are strings of text, numbers, dates, or times.



Example 4: Annotation Properties

In this example we show the input screen from the open source tool called Protégé which can be used to display certain portions of the Ontology. The Annotation tab displays metadata associated with Class, Object Property, Data Property, or Individual.

The prefixes rdfs, skos, and dc indicate the standard that defines this annotation. RDFS stands for Resource Description Framework Schema, SKOS stands for Simple Knowledge Organization System, and DC stands for Dublin Core. These represent various standards used to describe metadata about ontology classes, properties, and individuals.

CEDS' use of the annotation properties is as follows:

1. Rdfs:label – To list the CEDS element name or CEDS Option Description
2. Skos:prefLabel - To list the CEDS element name or CEDS Option Description
3. Dc:description – To list the CEDS element definition or the CEDS Option Definition
4. Skos:definition - To list the CEDS element definition or the CEDS Option Definition

5. Dc:creator – Presently only lists “Common Education Data Standard” but in the future as the ontology progresses and additional standards are able to be mergeed, it will list the organization responsible for that entity.
6. Dc:identifier – To list the unique identifier associated with the Class, Object Property, Data Property, or Individual. (In the example below, 001943162277 is the unique ID that represents “American Indian or Alaska Native” as an option of the “Race” element.
7. Skos:InScheme – indicates the related class
8. Skos:notation – the option set code or technical name for a class, object property, or data property.