# HED Ontology

*Release 0.0.1*

**HED Working Group**

**Dec 27, 2023**

# OVERVIEW:


**Links**

- PDF docs
- Source code

CHAPTER

ONE

# HED ONTOLOGY PROJECT

The HED ontology project is an effort to map HED schemas, which are hierarchical vocabularies designed for annotation, into formal ontologies. The goal is to provide additional metadata that is not accommodated in the HED schema files.

HED is an open research community effort; others interested are invited to participate and contribute. Visit the HED project homepage for links to the latest developments..

## 1.1 HED ontology development guide

### 1.1.1 Overview

#### 1.1.1.1 Naming conventions

##### 1.1.1.1.1 HED classes

- All hed classes have names starting with *Hed*.
- HED classes use upper camel case (all words start with a capital letter).
- Class names should be singular.

##### 1.1.1.1.2 HED properties

HED properties use lower camel case (all words except first start with a capital letter).

##### 1.1.1.1.3 HED instances of classes

In the HED schema HED tags start with a capital letter and individual words are hyphenated. These are converted upper camel case with underscores and hyphens removed:

**Example:** The HED tag *Sensory-event* is converted to *SensoryEvent*.

### 1.1.2 Annotation properties

| Property | Range | Target | Description |
|---|---|---|---|
| *library* | *xsd:string* | **HedHeader** | The name of the library schema if applicable. |
| *unmerged* | *xsd:boolean* | **HedHeader** | If true, schema is not merged with partner schema. |
| *version* | *xsd:boolean* | **HedHeader** | The semantic version of this schema. |
| *wikiSchemaSection* | *xsd:string* | **HedElement** | The name in Mediawiki format of the schema section this element belongs to. |
| *withStandard* | *xsd:boolean* | **HedHeader** | The version of the schema's standard schema partner. |
| *xmlSchemaSection* | *xsd:string* | **HedElement** | The name in XML format of the schema section this element belongs to. |

#### 1.1.2.1 Mapping rules

- The element (whether class, property, or instance) description (that appears in square brackets in the schema) is mapped to **rdfs:comment**.
- The name the element uses in the HED schema is mapped to **rdfs:label**.

##### 1.1.2.1.1 Schema attributes

Rules:

- Schema attributes have either *owl:DatatypeProperty* or *owl:ObjectProperty*.
- *owl:DatatypeProperty* are recognized by having *rdfs:subPropertyOf* value of either *hed:schemaAttributeDatatypeProperty* or *hed:schemaAttributeObjectProperty*,
- Schema attributes are mapped to owl:DatatypeProperty with if

#### 1.1.2.2 Schema sections

hed:HedSection The sections of the schema are specified by instances of the hed:HedSchemaSection class.

#### 1.1.2.3 Main classes

| Name | subclass | Description |
|---|---|---|
| ***HedElement*** | owl:Class | Superclass for specifying the structure of a HED schema. |
| ***HedHeader*** | owl:Class**HedElement** | Specifies the schema header. |
| ***HedNode*** | owl:Class**HedElement** | Specifies HED term or node elements. |
| ***HedPlaceholder*** | owl:Class**HedNode** | Specifies HED node placeholders. |
| ***HedSchemaAttribute*** | owl:Class**HedElement** | Defines attribute modifiers of other sections of the schema. |
| ***HedUnit*** | owl:Class**HedElement** | Specifies a HED unit . |
| ***HedUnitClass*** | owl:Class**HedElement** | Specifies HED units and unit modifiers. |
| ***HedUnitModifier*** | owl:Class**HedUnitClass** | Subclass of **HedUnitClass** defining unit modifiers. |
| ***HedValueClass*** | owl:Class**HedElement** | Defines the types of value classes. |

### 1.1.3 HedElement

**HedElement** is the superclass for all structure of the HED schema.

### 1.1.4 HedHeader

### 1.1.5 HedNode

**HedNode** is a class representing a HED tag or term. It is a subclass of **HedElement**.

**Rules for instances:**

- each HED tag or term is an instance of **HedNode**.
- every instance of **HedNode** has either the **hasHedParent** or the **isTopLevelTag** property.

### 1.1.6 HedPlaceholder

**HedPlaceholder** is a class representing the # nodes in the HED schema.

Each # in the schema corresponds to an instance of **HedPlaceholder**.

**Rules for instances of HedPlaceholder:**

- must have a **isTakesValue** property of true.
- may have one or more **hasValueClass** property items.
- may have one or more UnitClass instances.
- must have **isExtensionAllowed** property of False.
- must have **isTopLevelTag** property of False.
- must have a **hasHedParent** property.

### 1.1.7 HedSchemaAttribute

### 1.1.8 HedUnit

**HedUnit** specifies units.

**Rules for instances of HedUnit:**

- Must be an instance of **HedUnit**.
- Must have a **hasUnitClassType** property
- May have the **isSIUnit** property.
- May have a **isUnitSymbol** property.
- May have a **hasConversionFactorValue** property.

### 1.1.9 HedUnitClass

**HedUnitClass** specifies the types of HED units and unit modifiers.

**Rules for instances of HedUnitClass:**

- Each type of units such as *accelerationUnits* is an instance of **HedUnitClass**.
- An instance of **HedUnitClass** may have a **hasDefaultUnits** property.

### 1.1.10 HedUnitModifier

**HedUnitModifier** is a subclass of **HedUnitClass** that specifies unit modifiers.

**Rules for instances of HedUnitModifier:**

- Must be an instance of **HedUnitModifier**.
- Has either the **isSIUnitModifier** or **isSIUnitSymbolModifier** property.
- Must have the **hasConversionFactor** property.

### 1.1.11 HedValueClass

CHAPTER

TWO

# INDICES AND TABLES

- genindex
- search