

OWL 2 DL to Description Logic SROIQ to XBRL+DL

Description of Real World Idea ¹	XBRL	OWL 2 DL ²	Description Logics ³ (SROIQ)
A <i>classification system</i> ⁴ is a grouping of something based on some criteria. Represent information about categories of objects and how objects are interrelated with a goal of unambiguous interpretation.	Taxonomy	Ontology	State of the world
Notion of Open World Assumption and Closed World Assumption ⁵	Unspecified ⁶	OPEN or CLOSED WORLD ASSUMPTION	CLOSED WORLD ASSUMPTION
Notion of "multiplicity of sublanguages"	NO	OWL 2 Lite, OWL 2 DL, OWL 2 FULL	SROIQ, SROIN, RIQ, and various other description logics
Notion of "abstract"	Unsure; can define "abstract" using XML Schema, but seems like a different notion	NO	
Notion of "property inheritance"	Not defined	Formally Defined	Formally Defined
Notion of "real world things"	Instance, xbrl:item	Instance (RDF); rdf:about	Individual
Notion of "relationship"	To, Arcrole, From (provided by XLink)	Subject, Predicate, Object	Concept, Roles, Individuals

¹ Common to all possible models which can possibly exist in knowledgebase; goal is for information to be shared and exchanged without disputes as to precise meaning; lack of formality soon leads to arguments as to the meaning of language constructs. As expressive power increases, computational complexity increases and reasoning problems can result in unforeseen complexity blowups. Expressive power should be useful-yet-harmless. The goal is to properly balance the system with carefully chosen constructors and axioms such that typical applications with a requirement for reliable and efficient reasoning support. The best balance between expressiveness and complexity of reasoning depends on the intended application. An extreme case is when a knowledgebase is not satisfied by any interpretation. Such a knowledgebase is unsatisfied or inconsistent. In such vacuous cases where there are no interpretations, information is proven to be ambiguous. Such information clearly has no utility. Therefore, avoiding such cases during information representation is prudent. Such cases where no interpretations exist are referred to as "undecidable" or "unsatisfied" or "inconsistent". Rather than undecidable, a conclusion should always be reached as to the interpretation of a result. A result should be decidable. "Decidable" means that no interpretations that are not satisfied (unsatisfied or inconsistent) by at least one interpretation exists.

² OWL 2 Web Ontology Language, <http://www.w3.org/TR/2012/REC-owl2-primer-20121211/>

³ A *Description Logic Primer*, <http://arxiv.org/pdf/1201.4089.pdf>

⁴ Types of classification systems, <http://xbrl.squarespace.com/journal/2014/3/30/understanding-classification-systems.html>

⁵ Closed World Assumption "closes" the interpretation by assuming that every fact not explicitly stated to be true is actually false. This leaves no possibility for undecided interpretations. See, http://en.wikipedia.org/wiki/Closed-world_assumption and Open World Assumption, http://en.wikipedia.org/wiki/Open-world_assumption

⁶ CWA (closed world assumption) previously existed in XBRL but was removed.

Description of Real World Idea ¹	XBRL	OWL 2 DL ²	Description Logics ³ (SROIQ)
Notion of "data types" (includes restrictions on data types allowed by XML Schema, facets)	XML Schema Datatypes (includes restrictions on data types allowed by XML Schema, facets) ⁷ ; XBRL Data type registry ⁸	XML Schema Datatypes; <hasAge rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">51</hasAge>	DataPropertyAssertion(:hasAge :John "51"^^xsd:integer)
Notion of "unique identifier"	Name or ID? (What is the unique identifier; the name attribute or the ID attribute?)	Label	Name
PROPERTY: A property is a trait, quality, feature, attribute, or peculiarity which is used to define its possessor and is therefore dependent on the possessor (the thing which has the property). A property belongs to something. For example, the color of a ball belongs to and is therefore is dependent on (is a property of) the ball.	XML Schema Attribute	rdf:Description	ObjectPropertyAssertion(:hasWife :John :Mary)
CLASS: A set or category of things having some property in common and differentiated from other sets by kind, type, or quality. Group of real world things that go together for a specific reason. (Could be concrete or abstract)	Not defined	owl:Class	ClassAssertion(:Person :Mary)
CLASS HIERARCHY or SUB CLASS: Ability to organize classes into a hierarchy. Specialization of classes; broader terms versus narrower terms.	XBRL Definition relation general-special (not sure it is equivalent)	rdfs:subClassOf	SubClassOf(:Woman :Person)
CLASS EQUIVLANCE: Class "A" and class "B" have the exact same members. (Example, class <i>LiabilitiesAndPartnerCapital</i> and the class <i>LiabilitiesAndStockHolderEquity</i> are both equivalent to <i>LiabilitiesAndEquity</i> .)	Not defined	owl:equivalentClass	EquivalentClasses(:Person :Human)
INDIVIDUAL EQUIVALENCE: Individual "A" and individual "B" are the exact same real world thing. (Example, the individual (concept) <i>Equity</i> and the individual (concept) <i>NetAssets</i> are exactly the same thing.)	Not defined	owl:sameAs	SameIndividual(:John otherOnt:JohnBrown)
INDIVIDUAL NON-EQUIVALENCE: Individual "A" and individual "B" are NOT the same real world thing. (Example, the individual (concept) <i>Assets</i> and the individual (concept) <i>NetAssets</i> are NOT the same thing.)	Not defined	owl:differentFrom	DifferentIndividuals(:John :Bill)
CLASS (SET) DISJOINT: Things belonging to one class "A" cannot also belong to some other class "B". (Example, a member of the <i>Person</i> class set of things can never be a member of the <i>Country</i> class set of things.)	Not defined	owl:AllDisjointClasses	DisjointClasses(:Woman :Man)

⁷ XML Schema Data Types, <http://www.w3.org/TR/xmlschema11-2/>

⁸ XBRL International Data Type Registry, <http://www.xbrl.org/dtr/dtr.xml>

Description of Real World Idea ¹	XBRL	OWL 2 DL ²	Description Logics ³ (SROIQ)
CLASS (SET) COMPLIMENT: Things that are members of one class "A" are all the things that do not belong to the other class "B" (Example, a member of the class of LivingThings set of things is the entire set of things that do not belong to the DeadThings set of things.)	Not defined	owl:complementOf	ObjectComplementOf(:Parent)
CLASS (SET) INVERSE: A relationship of type "X" between "A" and "B" implies a relationship of type "Y" between "B" and "A". (Example, IF <starsIn><inverseOf><hasStar>; AND IF <MenInBlack><hasStar><WillSmith>; THEN <WillSmith><starsIn><MenInBlack>)	Not defined	owl:inverseOf	InverseObjectProperties(:hasParent :hasChild)
CLASS (SET) UNION: The members of set "C" include all the members of set "A" and all the members of set "B".	Not defined	owl:unionOf	ObjectUnionOf(:Mother :Father)
CLASS (SET) INTERSECTION: The members of set "C" include all the members of set "A" that are also members of set "B".	Not defined	owl:intersectionOf	ObjectIntersectionOf(:Woman :Parent)
INDIVIDUAL LINKED TO SELF: Individual is linked to itself	Not defined	owl:hasSelf	ObjectHasSelf(:loves)
SET MEMBERSHIP: Number of members of a set ⁹	Not defined	owl:maxQualifiedCardinality; owl:minQualifiedCardinality; owl:qualifiedCardinality; owl:cardinality	ObjectMaxCardinality(4 :hasChild :Parent); ObjectMinCardinality(2 :hasChild :Parent); ObjectExactCardinality(3 :hasChild :Parent)
INDIVIDUAL ENUMERATION: Enumeration of individuals	Not defined	owl:oneOf	ObjectOneOf(:Bill :John :Mary)
SYMMETRY: A relationship between "A" and "B" is also true between "B" and "A".	Not defined	owl:SymmetricProperty	SymmetricObjectProperty(:hasSpouse)
ASYMMETRY: Opposite of symmetry; if it connects A with B it never connects B with A.	Not defined	owl:AsymmetricProperty	AsymmetricObjectProperty(:hasChild)
TRANSITIVE: A relationship between "A" and "B" and between "B" and "C" is also true between "A" and "C".	Not defined	owl:TransitiveProperty	TransitiveObjectProperty(:hasAncestor)
REFLEXIVE: Such a property relates everything to itself. For example, note that everybody has himself as a relative.	Not defined	owl:ReflexiveProperty	ReflexiveObjectProperty(:hasRelative)
IRREFLEXIVE: No individual can be related to itself by such a role. For example, nobody can be his own parent.	Not defined	owl:IrreflexiveProperty	IrreflexiveObjectProperty(:parentOf)
INDIVIDUAL FUNCTIONAL OBJECT PROPERTY: Property can be related to at most one other individual. For example, every person can have only one husband; every individual can be linked by the hasHusband property to at most one other individual.	Not defined	owl:FunctionalProperty	FunctionalObjectProperty(:hasHusband)

⁹ Cardinality, <http://en.wikipedia.org/wiki/Cardinality>

Description of Real World Idea ¹	XBRL	OWL 2 DL ²	Description Logics ³ (SROIQ)
KEY: Each named instance of the class expression is uniquely identified by the set of values which these properties attain in relation to the instance. For example, the identification of a person by her social security number.	Not defined	owl:hasKey	HasKey(:Person () (:hasSSN))
DEFINED BY: Authoritative source of information.	Reference link	rdfs:isDefinedBy; rdfs:seeAlso	--NO SUPPORT--
INFORMATION ABOUT ONTOLOGY: General information about a topic is almost always gathered into an ontology that is then used by various applications.	XML Schema <documentation> element	owl:Ontology	Ontology(<http://example.com/owl/families>...)
COMMENT or ANNOTATION: Human readable definition or documentation of a concept.	Label link	rdfs:Comment	AnnotationAssertion(rdfs:comment :Person "Represents the set of all people.")
Units	Units, XBRL Units Registry ¹⁰	NO high level support, but can be represented	NO high level support, but can be represented
DIMENSIONAL MODEL: A characteristic describes a fact (a characteristic is a property of a fact). A characteristic or distinguishing aspect provides information necessary to describe a fact or distinguish one fact from another fact. A fact may have one or many distinguishing characteristics.	Aspects ¹¹ (XBRL dimensions, tuples)	NO high level support, but can be represented	NO high level support, but can be represented
DIMENSIONAL MODEL: A fact is reported. A fact defines a single, observable, reportable piece of information contained within a financial report, or fact value, contextualized for unambiguous interpretation or analysis by one or more distinguishing characteristics (properties of the fact). A fact value is one property of a fact. Every fact has exactly one fact value.	Simple facts (XBRL item), complex facts (tuples)	NO high level support, but can be represented	NO high level support, but can be represented
DIMENSIONAL MODEL: A hypercube is used to combine facts which go together for some specific reason. (AKA cube, matrix, table, lattice, array)	Hypercube, Network	Government Linked Data, Data Cube Vocabulary ¹²	NO high level support, but can be represented
COMPUTATIONS: A roll up computes a total from a set of concepts (stock or flow). This equation is: $A + B + n = \text{Total}$.	XBRL calculation relations	--NO SUPPORT--	
COMPUTATIONS: A roll forward reconciles a balance (stock) between two points in time (flow). This equation is: beginning balance + changes = ending balance.	XBRL Formula	--NO SUPPORT--	

¹⁰ XBRL International Units Registry, <http://www.xbrl.org/utr/utr.xml>

¹¹ XBRL International Abstract Model 2.0, <http://www.xbrl.org/specification/abstractmodel-primary/pwd-2012-06-06/abstractmodel-primary-pwd-2012-06-06.html>

¹² Data Cube Vocabulary, http://www.w3.org/2011/gld/wiki/Data_Cube_Vocabulary

Description of Real World Idea ¹	XBRL	OWL 2 DL ²	Description Logics ³ (SROIQ)
COMPUTATIONS: An adjustment reconciles an originally stated balance to a restated balance between two different report dates. This equation is: originally reported balance + adjustment = restated balance.	XBRL Formula	--NO SUPPORT--	
COMPUTATIONS: A member aggregation is a collection-member type whole-part relation. This equation is: Sum(concept).	XBRL Formula	--NO SUPPORT--	
Business rules	XBRL Formula	Rule Interchange Format (RIF)	
WHOLE-PART RELATIONS: A meronym denotes a constituent part of, or a member of something. (Whole-Part type relations ¹³)	XBRL definition relations, XBRL Link Role Registry (LRR)	Relations in general, but not specific relations types	
WHOLE-PART RELATIONS: HasPart (same as integral object-component)	Not defined	Can create	Can create
WHOLE-PART RELATIONS: IsPartOf (component-integral object)	Not defined	Can create	Can create
WHOLE-PART RELATIONS: member-collection; collection-member	Not defined	Can create	Can create
Simple and generic descriptions of electronic resources, Dublin Core ¹⁴	--NO SUPPORT--	Supports use of Dublin Core metadata	
Notion of enforcing "constraints" (like a relational database)	NO	NO	NO
IMPORT: Reuse of general information that is stored in one ontology in other ontologies (instead of requiring the copying of this information, allows the import of the contents of entire ontologies in other ontologies, using import statements:	xs:import; xs:include	owl:imports	Import(<http://example.org/otherOntologies/families.owl>)

¹³ A Taxonomy of Part-Whole Relations, <http://csjarchive.cogsci.rpi.edu/1987v11/i04/p0417p0444/MAIN.PDF>

¹⁴ Dublin Core, http://wiki.dublincore.org/index.php/User_Guide