

***Системи, основани на знания - зимен семестър,
2020/2021 учебна година***

***Тема 12:
Въведение в езика OWL***

Езикът OWL (Web Ontology Language) е създаден с цел да може да бъде използван за автоматична (компютърна, програмна) обработка на съдържанието на информационни ресурси, а не само за представяне на информация, предназначена за използване от хора.

В този смисъл OWL предоставя допълнителни (в сравнение с RDF/RDF Schema) възможности за „внасяне“ на семантика (значение) в описанията на съответните информационни ресурси.

OWL разширява (в сравнение с RDFS) речника, който може да бъде използван за описание на свойствата и класовете. В частност, възможно е да се описват релации между класове (например непресичащи се класове), кардиналност (например „точно един ...“), характеристики на свойствата (например симетричност) и др.

Подмножества на езика OWL

OWL включва три подмножества (езикови нива): OWL Lite, OWL DL и OWL Full. Тези подмножества са с нарастваща изразителна сила (в посочения ред) и са предназначени за използване от различни типове разработчици и крайни потребители.

- *OWL Lite* е полезен за тези потребители, които се нуждаят от средство за описание предимно на таксономии (класификационни йерархии) и прости ограничения.

- *OWL DL* е полезен за тези потребители, които имат потребност от максимална изразителна сила при запазване на изчислителната пълнота (за всички възможни заключения е гарантирано, че са изчислими).

OWL DL включва всички езикови конструкции, но те могат да бъдат използвани при определени ограничения (например, един клас може да бъде подклас на много класове, но не е допустимо един клас да бъде екземпляр на друг клас). Формална основа на OWL DL са т. нар. *description logics*.

- *OWL Full* е предназначен за потребители, които предпочитат максимална изразителна сила и синтактична свобода, без гаранции за изчислимост.

Кратък преглед на конструкциите на OWL Lite

OWL Lite поддържа част от възможностите на езика OWL и налага ограничения върху използването на някои езикови конструкции. OWL Lite може да се разглежда като подмножество на OWL DL, съответно OWL DL представлява подмножество на OWL Full.

Конструкции на OWL Lite, „наследени“ от RDF/RDFS

- ***rdfs:Class***: Всеки клас представлява множество от индивиди (индивидуални обекти), които са обединени от обстоятелството, че имат общи свойства. Класовете могат да бъдат организирани в йерархия с помощта на конструкцията *subClassOf*.

Съществува един максимално общ вграден клас, наречен Thing, който е суперклас на всички OWL класове. Съществува също един вграден клас, наречен Nothing, който няма екземпляри и е подклас на всички OWL класове.

- ***rdfs:subClassOf***: Позволява дефиниране на йерархии от класове, като специфицира даден клас като подклас на друг. Например, класът Person би могъл да бъде специфициран като подклас на класа Mammal.

По такъв начин всяка програма за логически извод (всеки ***reasoner***) ще може да направи заключение, че ако един обект е екземпляр на класа Person, то той същевременно е екземпляр на класа Mammal.

- ***rdf:Property***: Свойствата (properties) дефинират релации между индивидуални обекти или релации между индивидуални обекти и данни от определени типове. Примери за свойства: hasChild, hasRelative, hasSibling, hasAge. Класовете owl:ObjectProperty и owl:DatatypeProperty са подкласове на RDF класа rdf:Property.

- ***rdfs:subPropertyOf***: Позволява дефиниране на йерархии от свойства. Например, `hasSibling` може да се специфицира като подсвойство (subproperty) на `hasRelative`.

По този начин, ако един обект е свързан с друг чрез свойството `hasSibling`, то тези два обекта са свързани и чрез свойството `hasRelative` и това твърдение трябва да може да бъде изведено от всяка програма за логически извод.

- ***rdfs:domain***: Дефиниционната област (областта; domain) на едно свойство определя/ограничава множеството от индивиди, към които може да бъде прилагано това свойство.

- ***rdfs:range***: Определя/ограничава множеството от индивиди, които могат да бъдат стойност на съответното свойство.

- ***Individual:*** Индивидите (индивидуалните обекти) се дефинират като екземпляри на класове и могат да бъдат свързвани чрез свойства.

Равенство и неравенство в OWL Lite

- ***equivalentClass***: Позволява да се специфицира, че два класа са еквивалентни. Еквивалентните класове имат едни и същи екземпляри. Равенството може да се използва за създаване на класове – синоними.

- ***equivalentProperty***: Позволява да се специфицира, че две свойства са еквивалентни. Равенството може да се използва за създаване на свойства – синоними.

- ***sameAs***: Позволява да се специфицира, че два индивидуални обекта са еквивалентни. По такъв начин могат да бъдат създавани обекти, които имат много имена.

- ***differentFrom***: Позволява да се специфицира, че даден индивид е различен от други индивиди.
Възможността явно да се посочи, че определени индивиди са различни, е съществена, тъй като езиците от типа на OWL (и RDF) не предполагат, че индивидите имат само по едно име.

Характеристики на свойствата в OWL Lite

- ***inverseOf***: Позволява едно свойство да бъде специфицирано като обратно на друго.

- ***TransitiveProperty***: За всяко транзитивно свойство P е вярно, че ако двойките (x,y) и (y,z) са екземпляри на свойството P , то и двойката (x,z) също е екземпляр на P .

- ***SymmetricProperty***: Ако едно свойство P е симетрично, то винаги когато двойката (x, y) е екземпляр на P , двойката (y, x) също е екземпляр на P .

- ***FunctionalProperty***. Ако едно свойство е функционално, то това свойство има не повече от една стойност за всеки индивидуален обект.

Ограничения върху свойствата в OWL Lite

OWL Lite позволява да бъдат дефинирани ограничения върху това, как определени свойства могат да бъдат прилагани върху екземплярите на даден клас. Тези ограничения могат да бъдат специфицирани в контекста на конструкцията `owl:Restriction`.

- ***allValuesFrom***: Ограничението *allValuesFrom* се налага върху дадено свойство по отношение на някакъв клас. Означава, че по отношение на посочения клас свойството има асоциирано локално range ограничение.

Така, ако екземпляр на класа е свързан чрез това свойство с друг индивид, то за този индивид може да се изведе принадлежност към класа, посочен в локалното range ограничение.

Например за свойството `hasDaughter` може да е зададено ограничение *allValuesFrom* по отношение на класа `Person` със стойност `Woman` (ограничение, според което всички индивиди от класа `Person` могат да са свързани чрез свойството `hasDaughter` само с индивиди от клас `Woman`).

- ***someValuesFrom***: Ограничението *someValuesFrom* се налага върху дадено свойство по отношение на някакъв клас. Означава, че по отношение на този клас свойството има поне една стойност от определен тип (която е екземпляр на определен клас).

Допълнителни възможности на OWL DL и OWL Full

OWL DL и OWL Full използват един и същ речник, но в рамките на OWL DL се налагат множество ограничения върху използването на някои конструкции.

Например, OWL DL изисква сепарация на типовете (един клас не може да бъде същевременно индивидуален обект или свойство и едно свойство не може да бъде същевременно индивидуален обект или клас).

Освен това, OWL DL изисква всяко свойство да бъде или `ObjectProperty`, или `DatatypeProperty`: свойствата от първия тип са релации между екземпляри на класове, а свойствата от втория тип са релации между екземпляри на класове и литерали (данни) от типове на XML Schema.

- ***disjointWith***: Класовете могат да бъдат специфицирани като взаимно чужди (disjoint). Например, Man и Woman могат да бъдат специфицирани по този начин. Взаимно чуждите класове не могат да имат общи индивидуални обекти (обща екземпляри).

- ***hasValue***: Определя изискване даденото свойство да има за стойност конкретен индивид. Например екземплярите на класа `dutchCitizens` може да бъдат характеризирани като тези хора, за които свойството `Nationality` има стойност `theNetherlands`.

- ***oneOf***: Позволява дефиниране на класове чрез директно изброяване на техните екземпляри. Например класът `daysOfTheWeek` може да бъде описан чрез директно изброяване на принадлежащите му екземпляри `Sunday`, `Monday`, `Tuesday`, `Wednesday`, `Thursday`, `Friday`, `Saturday`.

- ***unionOf, complementOf, intersectionOf***. OWL DL и OWL Full позволяват задаване на произволни булеви комбинации от класове и ограничения (чрез описание на обединение, разлика/допълнение, сечение).

- ***minCardinality, maxCardinality, cardinality***: OWL DL и OWL Full позволяват задаване на произволни неотрицателни цели числа като стойности на тези ограничения.

- **Съставни класове:** За разлика от OWL Lite, който в много конструкции разрешава използване само на именувани класове (т.е. посочване само на имена на класове), OWL DL и OWL Full разрешават използване на съставни описания на класове. В допълнение OWL Full допуска класове да бъдат използвани като обекти/екземпляри на класове.