

**Consiglio Nazionale delle Ricerche
Istituto di Scienza e Tecnologie dell'Informazione
"A. Faedo" (ISTI)**

Laboratorio di Domotica

Progetto: Arianna-INDUSTRIA-2015

referente: TELECOM Italia spa

area obiettivo: presidio strategico dei mercati (soluzioni di logistica di filiera)

Ontologie Web

autore: Loredana Pillitteri

Indice

1	Introduzione.....	3
2	Ontologie.....	5
2.1	Sviluppare un'ontologia	7
2.1.1	Le fasi dello sviluppo – in dettaglio.....	8
2.2	I linguaggi per la costruzione di ontologie	10
2.3	I layers del web semantico	11
2.4	Il linguaggio RDF.....	13
2.5	Il linguaggio OWL.....	15
2.5.1.1	Un esempio	16
2.6	Protégé.....	17
3	Bibliografia.....	18

1 Introduzione

La globalizzazione ha portato la competizione in tutti i settori, persino nelle piccole e medie imprese di servizi, non è più prerogativa delle sole grandi imprese.

In un mercato in continuo regime di agitazione e di concorrenza, l'informazione diviene un bene sempre più prezioso e l'esigenza di amministrare ed organizzare l'informazione rilevante nel miglior modo possibile è sentita a qualsiasi livello aziendale.

Il web è un sistema per rappresentare informazioni, testuali e non, in un formato leggibile e sperabilmente comprensibile dalle persone!

Le persone interpretano il contenuto della pagina web mentre il navigatore non comprende il significato della pagina che sta visualizzando. Il motore di ricerca utilizza un archivio di parole chiave per restituire una pagina, utilizza sinonimi per ampliare la ricerca e fornire un elenco possibile di risposte ma non può fornire una risposta efficace!

L'idea del web-semantico nasce dalla necessità di estendere il web per favorire in aggiunta lo scambio di informazioni tra programmi (procedure) utilizzando una rappresentazione comprensibile (codificabile) dai computer .

La semantica è quella parte della linguistica che studia il significato delle parole/frasi mettendo in relazione l'espressione linguistica con il suo contenuto (cosa vuol dire)

Cio' implica un processo di interpretazione.

Quando un gruppo di agenti deve collaborare, siano essi persone o sistemi software, è necessario garantire che essi comprendano le richieste e le informazioni che ricevono.

L'interazione tra agenti dipende essenzialmente dall'adozione di una concettualizzazione, cioè una rappresentazione formale della realtà di una specifica situazione come percepita e organizzata da un agente e da un linguaggio comune.

T. Berners Lee, inventore del World Wide Web, scrive:

"Il Web fu disegnato come uno spazio di informazioni, con l'obiettivo di essere utile non solo per la comunicazione uomo-uomo, ma affinché anche le macchine potessero avere la possibilità di partecipare e dare il loro contributo. Uno dei maggiori ostacoli è stato il fatto che la maggior parte dell'informazione sul Web è disegnata per essere fruita dall'uomo (...), la struttura dei dati non è riconoscibile per un robot che naviga il Web. (...) L'approccio del Web Semantico, invece, sviluppa linguaggi per esprimere le informazioni in una forma accessibile e processabile da una macchina".

e racconta:

"Ho fatto un sogno riguardante il Web [...] ed è un sogno diviso in due parti. Nella prima parte, il Web diventa un mezzo di gran lunga più potente per favorire la collaborazione tra i popoli. Ho sempre immaginato lo spazio dell'informazione come una cosa a cui tutti abbiano accesso immediato e intuitivo, non solo per navigare ma anche per creare. Nella seconda parte del sogno, la collaborazione si allarga ai computer. Le macchine diventano capaci di analizzare tutti i dati sul Web, il contenuto, i link e le transazioni tra persone e computer. [...] i meccanismi quotidiani di commercio, burocrazia e vita saranno gestiti da macchine che parleranno a macchine, lasciando che gli uomini pensino soltanto a fornire l'ispirazione e l'intuito. Questo si concretizzerà introducendo una serie di progressi tecnici e di adeguamenti sociali attualmente in fase di sviluppo. [...] il Web sarà un luogo in cui l'improvvisazione dell'essere umano e il ragionamento della macchina coesisteranno in una miscela ideale e potente". [Tim-BL]

L'aumentare delle dimensioni della 'knowledge base' comporta all'utente un notevole incremento del tempo di ricerca e recupero dell'informazione comunemente indicata come 'information retrieval' ed ulteriore tempo per capire se le informazioni trovate possono essere pertinenti al problema da risolvere; un grande guadagno si avrebbe filtrando dalla mole di documenti a disposizione quelli che non interessano, 'industrializzando' il processo di ricerca per ottenere una risposta adeguata a determinate esigenze in tempi brevi.

Oggi la problematica si è spostata dall'esistenza dell'informazione, comunque molto probabile....., al reperimento della stessa e cioè, dando per scontato la presenza della risposta, il problema diventa raggiungerla estraendola da un contesto la cui dimensione cresce di giorno in giorno. Dunque l'approccio non automatico alla ricerca risulta inefficiente ed inefficace; le applicazioni software 'intelligenti' aiutano ad individuare catalogare ed utilizzare al meglio le informazioni disponibili a prescindere dal formato elettronico di archiviazione.

Appare subito chiara l'esigenza di avere uno strumento che permetta di abbattere i costi per l'information retrieval dedicando maggiori energie allo sviluppo di una soluzione al problema proposto dal cliente. Alternativamente alle tecnologie tradizionali l'uso della semantica, che opera sul significato delle parole e sulle relazioni tra esse, permette una maggiore accuratezza nell'individuazione dei documenti cercati.

2 Ontologie

rappresentare formalmente un insieme di concetti

Il processo di scambio di informazioni tra programmi può essere concretizzato rappresentando il modello di uno specifico ambito di conoscenza, vengono codificate le informazioni tramite i concetti attinenti e cioè tramite le classi e le relazioni tra esse, le regole, gli assiomi ed i vincoli specifici del dominio.

Tali concettualizzazioni sono comunemente definite '*Ontologie*'.

In filosofia l'Ontologia è una branca fondamentale della metafisica, è lo studio dell'essere in quanto tale, nonché delle sue categorie fondamentali (l'attribuzione di un predicato ad un soggetto).

In informatica l'Ontologia è una rappresentazione formale condivisa ed esplicita di una concettualizzazione di un dominio di interesse.

Un'ontologia è, in prima approssimazione, una tavola di categorie dove ogni tipo di entità è 'catturata' da qualche nodo in un albero gerarchico.

Più in dettaglio, si tratta di una *teoria assiomatica del primo ordine* (logica matematica) esprimibile in una logica descrittiva e cioè un sistema formale in cui si esprimono enunciati e conseguenze logiche in modo del tutto formale (regole di inferenza). Si tratta, in sintesi, di 'concettualizzare' un certo campo di conoscenza mediante una *rappresentazione astratta, semplificata e condivisa*.

Creare una ontologia vuol dire classificare le INFORMAZIONI raggruppandole in un dominio di conoscenza coerente e difatti l'ontologia è uno strumento sempre più diffuso tra gli sviluppatori Web per i vantaggi che offre nella condivisione delle informazioni.

Con un comune database relazionale, ad esempio un DB con le tabelle FELINO (es.record=leone), RETTILE, ERBIVORO, CARNIVORO, all'interrogazione "trova tutti i carnivori" il database non produrrà alcun risultato. Consideriamo invece l'Ontologia dello zoo, aggiungiamo l'assioma – *tutti i felini sono carnivori* – oltre alla classificazione degli animali secondo le famiglie ed alla stessa domanda la base di conoscenza restituirà l'elemento **leone** perchè, grazie alla capacità di ragionamento deduttiva, il sistema sarà in grado di ricavare l'informazione che il leone è un carnivoro (premessi che tutti i felini sono carnivori e che il leone è un felino).

Esistono differenti tipi di ontologia [ONTO-level] :

- *Top-level ontologies*: descrivono concetti molto generali o conoscenza di senso comune in maniera coerente e consistente, ma sono indipendenti dal dominio
- *Domain ontologies*: descrivono le categorie di una certa disciplina e sono legate ad uno specifico dominio di applicazione (esempio: medicina, chimica), non sono usate direttamente per costruire sistemi, quanto per costruire archivi di informazioni (knowledge base) ed aiutare così il lavoro cooperativo stabilendo un accordo sui termini di un dominio e del loro significato, comprensibile dai membri del team con diverso background culturale.
- *Task ontologies*: definiscono quali sono gli argomenti di un campo. Un campo può essere una disciplina, un settore industriale o una qualsiasi area della società che unifica molti domini di applicazione (esempio: diagnostica, vendite). Una disciplina ha quindi bisogno di diverse domain ontologies
- *Application ontologies*: descrivono la conoscenza proveniente da domain e task ontologies generalmente la loro specializzazione.

By [Tom Gruber](#)

in the *Encyclopedia of Database Systems*, Ling Liu and M. Tamer Özsu (Eds.), Springer-Verlag, 2009.[TomG-ONTO]

Sinonimi

computational ontology, semantic data model, ontological engineering

Alcune definizioni

- **Ontologia:** insieme di tecnologie per la modellazione di conoscenza
- **Modello Ontologico:** modello che definisce in modo esplicito le relazioni e la semantica delle entità di un dominio
- **Linguaggio Ontologico:** linguaggio per la descrizione di un modello ontologico

Una ontologia definisce i concetti, le relazioni e altre distinzioni di rilievo per modellare un dominio, si procede definendo un set di primitive che consente di modellare un dominio di conoscenza

Il set tipico è costituito da

classe o insieme

attributo o proprietà

relazione o relazioni tra classi

nel contesto dei database l'ontologia è vista sullo stesso livello di astrazione del data model.

2.1 Sviluppare un'ontologia

È utile chiedere agli esperti di dominio una descrizione in linguaggio naturale dei concetti salienti e dei collegamenti esistenti peculiari del dominio d'interesse.

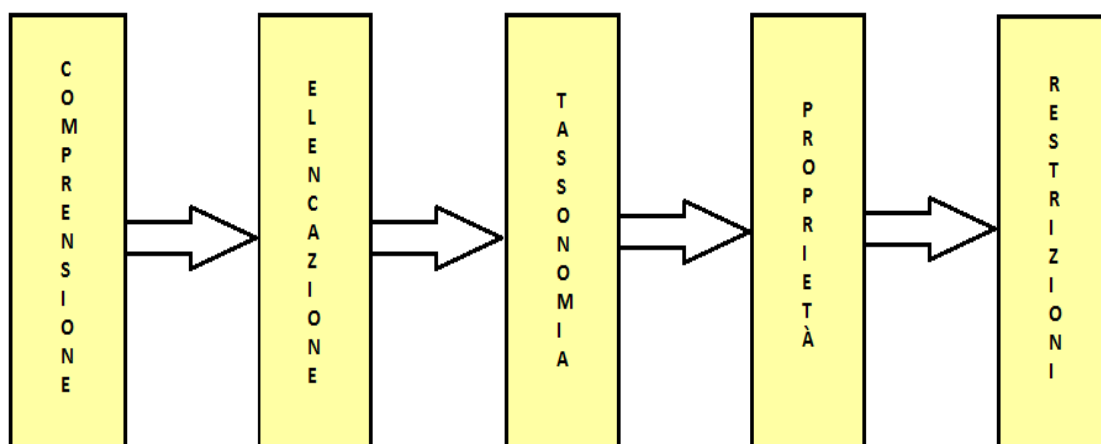
Il primo 'gradino' da realizzare per lo sviluppo di un'ontologia è quello della comprensione

Dopodichè dalle frasi in linguaggio naturale occorrerà individuare ed estrarre gli elementi (senza differenziare tra concetti, relazioni, o altro) enumerandoli
elencazione

Partendo dall'elenco prodotto si distingueranno ruoli e relazioni, si identificheranno allora le classi e le relazioni tra esse fornendo una strutturazione
tassonomia

Identificate le classi occorrerà definire le proprietà intrinseche della classe stessa e quelle legate ad altre classi che potranno dunque variare nel tempo
proprietà

Completando il processo di creazione di una ontologia occorrerà individuare le restrizioni di tipo e di cardinalità
restrizioni



2.1.1 Le fasi dello sviluppo – in dettaglio

Le fasi dello sviluppo di un'ontologia tipicamente sono

acquisire la conoscenza del dominio

In questa prima fase, collaborando con gli esperti del dominio, si cerca di raccogliere quante più informazioni possibili sul dominio di interesse e di comprendere i termini usati formalmente per poter descrivere le entità in maniera consistente. L'attenzione andrà posta sulle domande:

- quale dominio coprirà l'ontologia?
- qual è lo scopo dell'ontologia?
- a quali tipi di domande l'informazione espressa dall'ontologia potrà fornire risposte?
- chi userà e chi sarà il responsabile della manutenzione dell'ontologia?

considerare il riutilizzo di risorse esistenti

Effettuata la scelta del dominio, può essere utile controllare l'esistenza di risorse già presenti in rete da poter riutilizzare. Difatti l'idea di rifinire ed estendere risorse esistenti, quali glossari e dizionari dei termini e dei sinonimi, documenti, standard e altre ontologie è un vantaggio in termini di sviluppo dell'ontologia stessa e di tempo di realizzazione.

pianificare l'ontologia

Si inizia a progettare la struttura concettuale complessiva del dominio, identificandone i principali concetti e le loro proprietà, cercando le relazioni tra i vari concetti, eventualmente creandone di astratti, specificando quali di questi hanno delle istanze ecc.

E' utile chiedersi: quali sono i termini importanti? quali sono le proprietà?

vi sono tre passi fondamentali

- a) sviluppare un flat glossary: documentare ciascun termine con una definizione in linguaggio naturale, fornendo anche esempi appropriati in cui i nomi diventano oggetti o attori, ed i verbi si trasformano in relazioni o processi;
- b) elaborare un structured glossary: decomporre e/o specializzare i termini ed individuare gli attributi di un concetto;
- c) identificare tutte le relazioni concettuali fra gli oggetti.

organizzare e integrare l'ontologia

Aggiungendo concetti, relazioni ed entità, fino a raggiungere il livello di dettaglio necessario si cerca di integrare l'ontologia fino al raggiungimento degli obiettivi dell'ontologia stessa.

controllare il proprio lavoro

Sviluppata l'ontologia, occorre analizzarla per rilevare eventuali inconsistenze sintattiche, logiche e semantiche tra i suoi elementi. Spesso questa fase favorisce una classificazione automatica, che porta ad evidenziare nuovi concetti sulla base delle proprietà, delle entità e delle relazioni tra le classi.

consegnare l'ontologia

Terminato lo sviluppo di un'ontologia è necessario effettuare una verifica, da parte degli esperti del dominio, dopodichè si procede con la consegna del prodotto corredato di tutti i documenti relativi.

Un'ontologia è valida se ha le seguenti caratteristiche

- **completezza** prevedere tutte le distinzioni chiave
- **concisione** non fare assunzioni implicite e ridondanti
- **consistenza** non contenere definizioni contraddittorie. Una definizione è in contraddizione quando l'espressione delle altre definizioni dell'ontologia la contraddicono
- **coerenza** permettere la presenza di tutte e sole le relazioni consistenti con le definizioni dei concetti
- **riusabilità/scalabilità** poter estendere l'ontologia senza che l'estensione interessi le definizioni già presenti.

Non esiste tuttavia un modo univoco e corretto di modellare un dominio perché un'ontologia rappresenta una descrizione di una particolare realtà ed i concetti definiti in essa riflettono questa realtà; è altamente improbabile che un'ontologia possa contenere esaustivamente tutte le informazioni possibili sul dominio ed esprimere tutte le proprietà e distinzioni tra concetti nella gerarchia.

Appare evidente che è soprattutto l'esperienza che aiuta nel processo di costruzione di un'ontologia, è opportuno anche documentare ogni passo dello sviluppo, annotando eventuali problemi riscontrati e le soluzioni proposte, aiutando così gli utilizzatori e gli stessi progettisti per successivi cambiamenti.

Il lavoro della modellazione non è facile e farsi aiutare da uno strumento che fornisca una visualizzazione grafica dell'ontologia, suggerendo anche in maniera automatica concetti e relazioni, evitando di scrivere il codice a mano, può essere di fondamentale importanza.

2.2 I linguaggi per la costruzione di ontologie

Per essere utili, le ontologie devono essere espresse in una notazione concreta.

Un 'linguaggio per ontologie' è un linguaggio formale con cui un'ontologia viene costruita; per soddisfare le necessità delle ontologie esso deve possedere una serie di requisiti:

- estendere standard Web esistenti per semplificare il suo utilizzo
- essere facile da capire e da usare
- essere specificato in modo formale
- possedere un potere espressivo adeguato al dominio da descrivere.

Esistono diversi linguaggi, proprietari o basati su standard, per la definizione di ontologie; essi consentono di descrivere i metadati relativi alle risorse.

Il termine "risorsa" è usato fin dalle origini del web per indicare qualunque cosa reperibile in Internet mediante l'uso dei suoi protocolli e la generalità di questo termine ha incoraggiato un processo di generalizzazione sui metodi di accesso alle risorse stesse: da una prima idea (URL-**Uniform Resource Locator**) di limitarsi a **localizzare** una risorsa, si è passati all'idea di poterla *identificare* indipendentemente dalla sede (URI-**Universal Resource Identifier**) ed infine (RDF che descriveremo di seguito) a volerla **definire** mediante nessi semantici.

Una risorsa dovrebbe fornire informazioni su se stessa dunque parliamo di metadati. (*)

I metadati dovrebbero essere "*scritti*" in una forma leggibile universalmente dalle macchine. 'Aggiungere' la semantica ai contenuti WEB richiede la creazione di linguaggi e tecnologie in grado di estrarre significati da informazioni, e, cosa ancor più complessa, esprimere dati e regole per ragionamenti.

(*) Etimologicamente, il termine richiama l'idea di *dati di secondo livello*, dati utilizzati per descrivere e classificare altri dati.

L'esempio tipico, più comunemente utilizzato per spiegare il concetto, è il catalogo di una biblioteca. Gli scaffali di una biblioteca raccolgono libri (dati di primo livello o 'informazione primaria'), mentre il catalogo raccoglie schede che descrivono i libri, sulla base di una serie di caratteristiche prefissate: nome dell'autore, titolo, anno di pubblicazione, numero di pagine, posizione negli scaffali-biblioteca e così via..... Le schede di un catalogo contengono dunque dei dati di secondo livello, dei metadati appunto, che descrivono l'informazione primaria e aiutano a gestirla e reperirla. Dal canto suo, la biblioteca comprende sia i libri sia il catalogo: una buona raccolta di risorse informative, infatti, comprende sempre anche informazione secondaria, metadati, che permettono di descrivere, organizzare e reperire l'informazione primaria.

2.3 I layers del web semantico

Si può considerare il web semantico come una struttura articolata in diversi livelli, strati o *layers*, tutti necessari alla sua piena implementazione

livello 1 è il livello della definizione univoca delle risorse e si basa da un lato su UNICODE, che si occupa di rappresentare le lingue esistenti (i linguaggi del w3c si basano su UTF-8) dall'altro sullo standard URI (Uniform Resource Identifiers) l'identificazione avviene tramite il formato

```
<scheme name> : <hierarchical part> [ ? <query> ] [ # <fragment> ]
```

Gli URI sono estendibili, completi, stampabili.

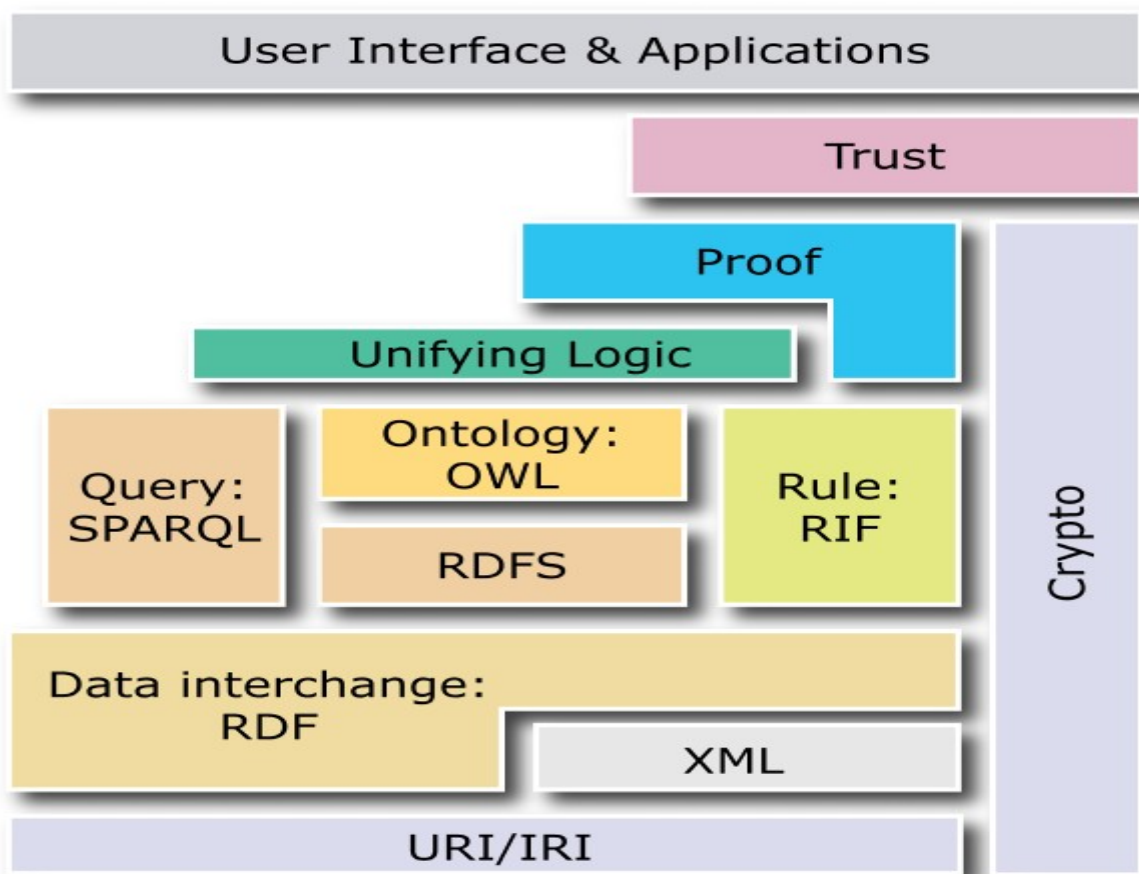
livello 2 protagonista XML (eXtensible Markup Language), che gioca un ruolo di base con i namespace (un prefisso identificativo anteposto al nome dell'elemento del marcatore interessato per qualificarlo in modo univoco) e gli XML Schema.

livello 3 livello di RDF (Resource Description Framework) e RDF Schema, che costituiscono il linguaggio per descrivere le risorse e i loro tipi. Il primo serve a descrivere affermazioni "machine-processable" della forma soggetto-predicato-complemento; il secondo crea vocaboli per i documenti RDF, esprime i vincoli sul dominio dei termini e delle proprietà, lega domini e codomini come nella programmazione ad oggetti.

livello 4 è lo strato ontologico, il cuore del web semantico. Attraverso una ontologia si possono descrivere le relazioni concettuali, come ad esempio le relazioni in DAML: "InverseOf" (forma passivante) e "EquivalentTo" (sinonimi) nonché le proprietà dei vocaboli.

livello 5 è quello in cui deve operare la logica, nel senso che si può acquisire nuova conoscenza a partire dalle asserzioni reperibili sul web. Per ovviare alla mancanza di interoperabilità dei sistemi deduttivi è stato suggerito di definire un linguaggio universale capace di rappresentare le dimostrazioni (ad es. PROOF).

livello 6 è il livello in cui linguaggi come PROOF in combinazione con la tecnologia della firma digitale, portano alla creazione di reti di fiducia e sfiducia. In effetti, la crittografia a chiave pubblica appare come un tassello essenziale del grande progetto del web semantico, giacché offre la possibilità di introdurre dei meccanismi affidabili per la verifica della provenienza di dati, metadati, ontologie e deduzioni.



http://it.wikipedia.org/wiki/Web_semantico#OWL

2.4 Il linguaggio RDF

RDF-Resource Description Framework <http://www.w3.org/RDF/>

una suite di W3C Recommendations, published in 2004.

Il livello infrastrutturale delle specifiche relative all'ambito del web semantico viene definito *Resource Definition Framework* (RDF) come dire inquadramento per la definizione di risorse.

RDF Resource Description Framework [W3C-RDF] è un framework per la descrizione della conoscenza nel web. È stato specificatamente creato per la descrizione dei metadati relativi alle risorse.

Questo framework è alla base del web semantico e permette la condivisione di conoscenza sul web. Il modello di dati RDF è formato da risorse, proprietà e valori.

Le proprietà sono delle relazioni che legano tra loro risorse e valori e sono anch'esse identificate da URI.

Un valore, invece, o è una risorsa o è un tipo di dato primitivo.

L'unità base per rappresentare un'informazione in RDF è lo statement., uno statement è una tripla del tipo: Soggetto – Predicato – Oggetto

dove il Soggetto è una risorsa, il Predicato è una proprietà e l'Oggetto è un valore

dunque si esprimono le informazioni usando la logica dei predicati mediante **asserzioni** (*statement*).

Il data model RDF permette di definire un modello semplice per descrivere le relazioni tra le risorse, in termini di proprietà identificate da un nome e relativi valori, tuttavia non fornisce alcun meccanismo per dichiarare queste proprietà, né per definire le relazioni tra queste ed altre.

RDF è lo strumento base per la codifica, lo scambio e il riutilizzo di metadati strutturati, e consente l'interoperabilità tra applicazioni che si scambiano sul Web informazioni machine-understandable.

I settori nei quali RDF può essere utilizzato e portare vantaggi sono i più disparati

- descrizione del contenuto di un sito Web, o di una pagina, o di una biblioteca digitale;
- implementazione di intelligent software agent, per lo scambio di conoscenza e un utilizzo migliore delle risorse Web;
- classificazione del contenuto, per applicare criteri di selezione;
- descrizione di un insieme di pagine, che rappresentano un singolo documento logico;
- stabilire i criteri di proprietà intellettuale delle singole pagine;
- esprimere criteri di privacy preference degli utenti e le privacy policies di un sito Web;
- con il meccanismo della digital signature, contribuire alla creazione del Web of Trust, per le applicazioni nel commercio elettronico, la cooperazione, etc....

Il Resource Description Framework dunque non descrive la semantica ma fornisce una base comune per poterla esprimere, permettendo di definire la semantica dei tag XML.

RDF è costituito da due componenti:

- RDF Model and Syntax che definisce il data model RDF e la sua codifica XML
- RDF Schema che permette di definire specifici vocabolari per i metadati.

La sintassi di base RDF prevede di definire i nessi concettuali attraverso la definizione di **predicati** (o **proprietà**) che mettono in relazione un **soggetto** ad un **oggetto**.



L'RDF fornisce dunque lo strumento sintattico per costruire terne relazionali.

Ciascuno dei partecipanti a tali nessi concettuali è di fatto un URI e ciascuna URI può partecipare ad altre relazioni, anche con ruoli diversi. Si capisce facilmente che mettere in partecipazione delle URI in sistemi ternari soggetto - predicato- oggetto equivale a coinvolgere le risorse in rete in un sistema contestualizzato e passibile di esporre un **significato**.

2.5 Il linguaggio OWL

OWL Web Ontology Language standard emergente delle ontologie Web
[online] <http://www.w3.org/TR/owl2-overview/>

L'RDF, pur prevedendo ulteriori dettagli sintattici ben più sottili e potenti della sola definizione di terne non è sufficiente ad implementare funzionalità semantiche nelle applicazioni web. Per ottenere una completa abilitazione semantica è necessario aggiungere vincoli particolari alla sintassi RDF: è questo lo scopo dell'OWL (*Ontology Web Language*) un linguaggio della famiglia delle Logiche Descrittive espresso in sintassi XML. Sviluppato come passo successivo di RDF e RDFS, così come precedenti progetti di linguaggi per ontologie, fra cui Ontology Inference Layer (OIL) e DAML, rappresenta l'attuale standard per ontologie in ambienti Web.

L'obiettivo dell'OWL non è naturalmente solo quello di consentire l'attribuzione di significati alle risorse (basterebbe un efficiente sistema di vocabolario informatizzato), ma di rendere anche tali significati **computabili**, ossia di consentire a meccanismi automatici (in particolare a computer o reti di computer) di valutare **inferenze** su tali significati.

Secondo una consuetudine ormai consolidata nelle tecnologie informatiche a seguito dell'universale adozione del principio *Object Oriented*, inoltre, le risorse in ambito RDF e OWL sono inquadrare come **classi** (tipo concettuale e generalmente astratto di oggetti) e **istanze** (oggetti concreti).

Un insieme di definizioni che rispetti la sintassi OWL prende il nome di **ontologia**, per amore di una sorta di metonimica brevità comune nel gergo usato per i formalismi informatici e matematici in generale. Le specifiche RDF, infine, consentono di concepire architetture aperte ed è facilmente comprensibile come l'uso estensivo degli URI consenta a qualunque ontologia di usare altre ontologie già definite altrove.

L'implementazione delle relazioni RDF e OWL avviene come di consueto in XML. Esempi concreti si possono reperire in [RDF Primer](#), presente nella sezione della W3C dedicata alla [Semantic Web Activity](#).

OWL è stato rilasciato in tre diverse versioni con complessità e potere espressivo crescenti:

OWL-Lite è la versione sintatticamente più semplice, attraverso cui è possibile definire gerarchie di classi e vincoli poco complessi

OWL-DL è la versione intermedia, così chiamato per la sua corrispondenza con le Logiche Descrittive

Un linguaggio descrittivo è più semplice di un linguaggio del primo ordine perché contiene solo concetti atomici, ruoli ('roles' in inglese) e nomi di oggetti. Un concetto atomico ad esempio un nome comune come 'padre', 'moglie', etc....; un ruolo è una relazione binaria e un nome di oggetto rappresenta appunto un oggetto singolo. OWL-DL offre un potere espressivo più elevato e mantiene la completezza computazionale (tutte le conclusioni sono computabili) e la decidibilità (tutte le computazioni si concludono in un tempo finito).

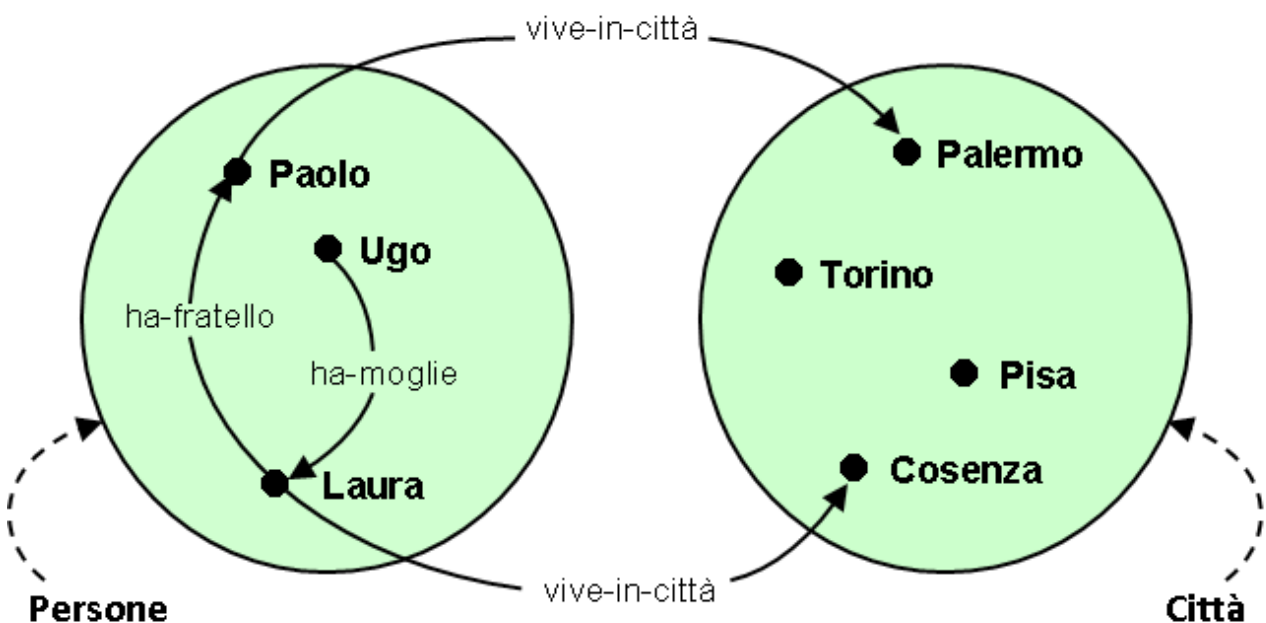
OWL- Full offre massima espressività senza alcuna garanzia circa completezza e decidibilità.

Ogni versione del linguaggio include ed estende la precedente, dunque un'ontologia OWL-Lite è sempre valida anche in OWL-DL, un'ontologia OWL-DL è sempre valida anche in OWL-Full .

I componenti principali di un'ontologia OWL sono tre: individui, proprietà e classi.

Gli individui rappresentano gli oggetti nel dominio di interesse, le proprietà sono relazioni binarie (ovvero che collegano due oggetti per volta) tra individui, le classi sono gruppi di individui.

2.5.1.1 Un esempio



un esempio dove 7 individui (Paolo, Palermo, Torino, Ugo, Pisa, Cosenza e Laura) sono raggruppati in 2 classi (Città e Persone) e relazionati attraverso tre tipi di proprietà (ha-fratello, ha-moglie e vive-in-città).

L'individuo è rappresentato tramite un piccolo tondo pieno, la classe con un cerchio vuoto e la proprietà con un arco direzionato.

Un'ontologia costruita sull'esempio ci direbbe che Paolo, Ugo e Laura sono persone mentre Palermo, Pisa, Torino e Cosenza sono Città. Ugo ha per moglie Laura che vive a Cosenza.

Laura, a sua volta, ha un fratello che si chiama Paolo e vive a Palermo. Le classi OWL possono essere organizzate in gerarchie di superclassi e sottoclassi dette tassonomie. Nell'esempio precedente, dalla classe Persone è possibile derivare la sotto-classe Uomini e la sotto-classe Donne. Dire che Uomini è sotto-classe di Persone significa affermare che tutti gli Uomini sono Persone.

2.6 Protégé

Il software più usato per la creazione di un'ontologia è **Protégé**. [Protégé] è una piattaforma open-source sviluppata dal Stanford Center for Biomedical Informatics Research presso Stanford University School of Medicine che fornisce, a una crescente comunità di utenti, una suite di strumenti per realizzare modelli che coprono vari domini: dal campo medico (per modellare la diffusione di malattie) al campo militare (per la gestione delle centrali nucleari). E' scaricabile gratuitamente dal sito <http://protege.stanford.edu/> e fornisce un ambiente grafico e interattivo per la progettazione delle ontologie ed un ambiente di sviluppo concettuale. Aiuta gli esperti del dominio a realizzare applicazioni per la gestione delle informazioni. Gli sviluppatori di ontologie possono accedere ad informazioni rilevanti in maniera semplice e veloce ogni volta che ne hanno bisogno e possono usare strumenti di manipolazione diretta per navigare tra le ontologie; inoltre i comandi utilizzabili per le gerarchie (tree controls, alberi di ontologie) consentono una veloce e semplice navigazione tra le gerarchie di classi.

La piattaforma Protégé supporta due modi principali di modellazione di ontologie:

- a) il Protégé- Frames editor che consente agli utenti di costruire e popolare ontologie che sono basate su 'frame', in conformità con il protocollo Open Knowledge Base Connectivity (OKBC). In questo modello, un'ontologia è costituita da un insieme di classi organizzate in gerarchia, rappresentanti un insieme di concetti. Le classi sono caratterizzate da proprietà (slot) e relazioni
- b) il Protégé-OWL editor che consente agli utenti di creare ontologie per il Semantic Web, in particolare nel W3C Web Ontology Language (OWL). Un'ontologia OWL può includere descrizioni di classi, di proprietà e le loro istanze.

Oltre alla presenza di una semplice interfaccia, Protégé fornisce

- supporto per le classi e le gerarchie di classi con molti legami di ereditarietà
- svariati template slot pronti per l'uso
- specifiche degli attributi degli slot, che includono valori consentiti, restrizioni sulla cardinalità, valori predefiniti
- metaclassi (classi per gestire le classi dei domini) e gerarchie di metaclassi.

Altre due caratteristiche distinguono Protégé dai molti ambienti di sviluppo per le ontologie sono la **scalabilità** e l'**estendibilità**.

Il sistema è modulare, la sua architettura, basata su componenti, semplifica l'aggiunta di nuove funzionalità, creando plug-in appropriati o utilizzandone di già esistenti; la Protégé Plugin Library raccoglie plug-in creati da sviluppatori di tutto il mondo.

I plug-in più usati sono quelli che forniscono capacità di visualizzazione avanzata, controllo di versione e così via. Un esempio è OntoViz che visualizza un'ontologia come un grafico sfruttando una libreria open source ottimizzata per la visualizzazione grafica; le classi e le istanze sono rappresentate come nodi, mentre le relazioni sono visualizzate come archi orientati. Sia i nodi sia gli archi sono classificati e disposti in un modo tale da ridurre al minimo le sovrapposizioni, a scapito della dimensione del grafico. Altro esempio per la visualizzazione di ontologie, questa volta in 3D, è Ontosphere3d [Ontosphere], sviluppato dal gruppo E-Lite del Politecnico di Torino.

3 Bibliografia

[Ontosphere] *Ontosphere3D*, <http://ontosphere3d.sourceforge.net/>

[ONTO-level] Berardina Nadja De Carolis, *Ontologie e rappresentazione della conoscenza*
<http://www.di.uniba.it/~nadja/sysag/ontologie.pdf>;

[Protégé] Protégé, open source ontology editor and knowledge-base framework, Homepage
<http://protege.stanford.edu/>;

[Tim-BL] http://it.wikiquote.org/wiki/Tim_Berners-Lee

[TomG-ONTO] <http://tomgruber.org/writing/ontology-definition-2007.htm>
by Tom Gruber in the *Encyclopedia of Database Systems*, Ling Liu and M. Tamer Özsu (Eds.),
Springer-Verlag, 2009.

[W3C-RDF] RDF, Resource Description Framework Homepage, <http://www.w3.org/RDF/>

approfondimenti

(OWL) Web Ontology Language Homepage, <http://www.w3.org/TR/owl-features/>

(RDFS) *RDF-Schema Homepage*, <http://www.w3.org/2001/sw/wiki/RDFS>

(W3C-OWL) *Web Ontology Language overview, 2009*