

I metadati e il Semantic Web

TW

Fabio Vitali



Indice degli argomenti

Oggi parliamo di:

- ◆ La necessità del Semantic Web
- ◆ L'organizzazione delle informazioni
- ◆ La struttura del Semantic Web

TW

Perché Semantic Web?

TW

- La sostanziale inconfontabilità del sapere
- Il concetto di semantica in un ambito puramente sintattico



Dati, informazioni, conoscenza (1)

I dati sono strutture discrete, precise, quantificate, non ambigue

- ◆ “6 aprile 2011”, “Piazzale Medaglie d’Oro 1”, “Nel mezzo del cammin di nostra vita”, ecc.
- ◆ I computer sono bravissimi a memorizzare e trovare dati.

Le informazioni sono la qualificazione di questi dati in maniera operabile da applicazioni

- ◆ “6 aprile 2011” è una data, e tra le operazioni c’è il calcolare quanti giorni mancano da oggi.
- ◆ “Piazzale Medaglie d’Oro 1” è un indirizzo, e tra le operazioni c’è il calcolare la strada per arrivarci da qui.
- ◆ “Nel mezzo del cammin di nostra vita” è un verso poetico, e tra le operazioni c’è il visualizzare il testo su una riga sola sullo schermo di un computer.
- ◆ La qualificazione dei dati come informazioni è indispensabile per la esecuzione di operazioni su di essi.

Dati, informazioni, conoscenza (2)

La conoscenza è fornito dal collegare le informazioni in maniera sistematica attribuendo loro ruoli e scopi.

- ◆ “6 aprile 2011” è la “data di inizio” di un “processo giudiziario” il cui “nome” è “Processo Ruby” e che vede tra gli “attori” anche la “persona” il cui “nome” è “Silvio Berlusconi” e il cui “ruolo” è “Presidente del Consiglio”.
- ◆ “Piazzale Medaglie d’Oro 1” è l’ “indirizzo” di un “luogo” chiamato “Stazione Centrale”, il cui ruolo è “stazione ferroviaria” e le cui “coordinate geografiche” sono “44.5056,11.3433”
- ◆ “Nel mezzo del cammin di nostra vita” è un “verso” di un “poema” il cui “titolo” è “Divina Commedia” e il cui “autore” è la “persona” il cui “nome” è “Dante Alighieri”.

Non basta: la conoscenza può essere organizzata fornendo caratterizzazione e vincoli astratti:

- ◆ Un “processo” è un tipo di “evento” che si svolge in “luoghi” di tipo “aule di tribunale”, a cui partecipano “persone” nel ruolo di “imputati”, “persone” nel ruolo di “avvocati”, “persone” nel ruolo di “pubblici ministeri” e “persone” nel ruolo di “giudici”.
- ◆ I “treni” sono “mezzi di trasporto” che “fermano” presso “stazioni ferroviarie” ad “orari”.
- ◆ I “poeti” sono “persone” che sono “autori” di un “poema”

L'inconfrontabilità del sapere nella forma testuale

La difficoltà di questa organizzazione ha molto a che fare con alcuni problemi specifici:

- ◆ Differenza tra termini usati dall'autore e dal lettore
 - ✦ Io cerco "mal di testa", i documenti contengono "emicrania"
- ◆ Molteplicità di termini usati per stile o abitudine
 - ✦ Cavaliere, Presidente del Consiglio dei Ministri, Silvio Berlusconi, papi, leader del PdL, Berluskaizer, imputato, primo ministro, ecc.
- ◆ Ambiguità intrinseca di alcuni termini
 - ✦ "L'importanza della pesca nell'economia della provincia": a Parma (coltivano frutta) interpretano diversamente che a Ravenna (c'è un porto).

Cosa capisce un computer

```
<frogle>  
  <plep>Grntt</plep>  
  <glep>Pfft</glep>  
  <trontig>  
    <stup>Pitt</stup>  
    <plit>q3978r29</plit>  
    <thra>Lippray Glurt</thra>  
  </trontig>  
</frogle>
```

Ha esattamente lo stesso significato, per uno strumento informatico, di:

```
<persona>  
  <nome>Silvio</nome>  
  <cognome>Berlusconi</cognome>  
  <imputato>  
    <tribunale>Milano</tribunale>  
    <processo>12345</processo>  
    <giudice>Giulia Turri</giudice>  
  </imputato>  
</persona>
```

Dov'è il significato?

- Non nei dati
- Non nel markup (i tag)
- Non nel documento che specifica il vocabolario ristretto e i suoi vincoli (il DTD o XML Schema)
- Non nello strumento XML

Ma:

- nell'applicazione che gestisce il contenuto del documento XML...
- ... e nella mente dell'essere umano che scrive o legge il contenuto del documento XML

XSLT per confrontare il significato?

Con strumenti XML come XSLT possiamo convertire gli elementi in strutture paragonabili. Ad esempio possiamo convertire:

- ◆ <frogle> e <persona> in tabelle,
- ◆ <plep>, <grep>, <nome> e <cognome> in celle di tabella,
- ◆ <trontig> e <tribunale> in paragrafi.

Questo ci permette di dire che persona o frogle sono tabelle? Certo che no.

Questo ci permette di dire che persona e frogle sono la stessa cosa? Certo che no.

XSLT serve solo per esprimere equivalenze locali (cioè, dipendenti dall'applicazione) tra termini di due vocabolari ristretti.

La risposta: il semantic web

L'arma definitiva per l'appassionato fan della dichiaratività
Ancora più astratto, ancora più sintattico, ancora più privo di significati e comportamenti predefiniti
Nel semantic web l'universo è formato da affermazioni su qualche classe o proprietà del dominio del discorso. Queste affermazioni, ovviamente, non hanno significato predefinito, e richiedono strumenti software o cervelli per fornirsene.
RDF è un meccanismo puramente sintattico per esprimere affermazioni, OWL è un meccanismo puramente sintattico per confrontarle e generarne (inferirne) di nuove.

La semantica in RDF

Con RDF (Resource Description Framework), possiamo fare affermazioni tipo:

- ◆ L'autore della Divina Commedia è Dante Alighieri
- ◆ La stazione ferroviaria si trova in "Piazzale Medaglie D'Oro 1"
- ◆ L'imputato del processo è Silvio Berlusconi

OWL per confrontare e inferire

OWL (Web Ontology Language) permette di fornire affermazioni che confrontano affermazioni di un livello inferiore e possono generare nuove affermazioni dello stesso livello.

Con OWL, è possibile fare affermazioni come:

- ◆ Esiste il concetto (classe) “Essere intelligente” e ogni affermazione fatta su un person o un frogle condividono questo concetto (*appartengono alla stessa classe*)
- ◆ Esiste una sottoclasse “Umano” e una sottoclasse “Marziano” della classe “Essere intelligente”
- ◆ Address e name sono caratteristiche (*proprietà*) della sottoclasse “Umano” e trontig e plep sono caratteristiche (*proprietà*) della sottoclasse “Marziano”
- ◆ Affermazioni su un address per un “Umano” sono equivalenti ad affermazioni su un trontig su un “Marziano”, così come affermazioni su un name sono equivalenti ad un plep, e così via.

Con una collezione sufficientemente ricca di queste affermazioni su classi e proprietà, è possibile a questo punto confrontare esempi di frogle e di person.

Ci siamo dotati di un modello di riferimento che ci permette di inferire nuove affermazioni su Umani e Marziani, e forse anche su Esseri Intelligenti.

Una parentesi: l'organizzazione delle informazioni

Per capire a cosa serve il Semantic Web, dobbiamo prima introdurre alcuni termini:

- ◆ Metadati e metainformazioni
- ◆ Vocabolario controllato
- ◆ Tassonomia
- ◆ Thesaurus (o tesoro)
- ◆ Classificazione a faccette
- ◆ Ontologia
- ◆ Folksonomia

Metadati e metainformazioni

Ogni dato o informazione che parli di un dato o di una risorsa informativa. Utile per classificare, descrivere, ricercare, confrontare, sfruttare in maniera sofisticata la risorsa.

In quanto metadato, è un'affermazione che parla di una risorsa. Quindi il riferimento alla risorsa è sostanzialmente obbligato.

Es.: L'autore di questa dispensa è Fabio Vitali

L'autore del metadato può essere l'autore della risorsa, e il metadato posizionato nella risorsa stessa (e allora parliamo di proprietà riflessive della risorsa), o esterne per autore e posizione alla risorsa.

Vocabolario controllato

Anche: linguaggio di indicizzazione

Alcuni metadati (ad esempio l'identificazione dell'autore di una risorsa) richiedono valori da un insieme aperto (tutti i nomi di persone al mondo)

Altri metadati richiedono che i valori siano compresi in un insieme di valori precisi:

- ◆ Dotati di significato e di applicabilità
- ◆ Non ridondanti
- ◆ Non ambigui
- ◆ Completi rispetto al dominio dei valori possibili

Tassonomia (1)

Termine inventato da Carlo Linneo nel XVIII secolo per la classificazione degli esseri viventi.

Crea una gerarchia tra i termini di un vocabolario controllato, in grado di esplicitare relazioni di specificità o generalità tra i termini.

Ad esempio:

- ◆ Romeo è un soriano
- ◆ Un soriano è un gatto, un gatto è un felino, un felino è un mammifero, un mammifero è un animale.

La tassonomia non cambia il metadato, né i valori possibili (che sono sempre appartenenti al vocabolario controllato),

Fornisce un ordine e una organizzazione ai termini del vocabolario controllato.

Tassonomia (2)

Alcune (molte) tassonomie introducono termini non istanziabili - cioè non usabili come valori di metadati, unicamente come raccordo tra i valori possibili.

- ◆ non esiste nessun animale che sia un mammifero senza essere anche un felino, o un canide, o un primate, ecc.

La tassonomia è un'operazione linguistica, non scientifica:

- ◆ fa parte del modello della realtà, non della realtà.
- ◆ Serve agli umani per comprendere e usare la realtà, ma non ha nessuna necessaria attinenza con la realtà vera.

I tesauri (o thesauri)

In breve, un tesauo è una tassonomia a cui si aggiungono relazioni di pari livello tra termini.

Definizione di tesauo (ISO 2788-1986) «il thesaurus è il vocabolario di un "linguaggio di indicizzazione" controllato, organizzato in maniera formale, in maniera cioè da rendere esplicite le relazioni "a priori" fra i concetti»

Il tesauo permette di

- ◆ trovare un punto di incontro tra lessico dell'autore e lessico del ricercatore,
- ◆ Proporre una relazione biunivoca tra termine e concetto, così da ottenere *univocità semantica*:
- ◆ un termine per ogni concetto, un concetto per ogni termine.

I tesauri (2)

L'univocità semantica elimina i problemi connessi con l'uso del linguaggio naturale

- ◆ ridondanze, ambiguità, polisemie, omonimie, omografie
- ◆ queste caratteristiche garantiscono ricchezza ed espressività al linguaggio naturale, ma rendono difficile l'organizzazione funzionale dei motori di ricerca.

I tesauri generalizzano la gerarchia tra termini della tassonomia in un generico insieme di relazioni tra termini, alcuni gerarchici, altri no.

- ◆ Relazione gerarchica
- ◆ Relazione preferenziale o sinonimica
- ◆ Relazione associativa

Relazioni tra termini nei tesauri

Relazione gerarchica

- ◆ Relazione di subordinazione all'interno di uno stesso albero gerarchico.
- ◆ Es.: matematica/geometria, felini/gatti, veicoli/automobili

Relazione preferenziale o sinonimica

- ◆ Identifica tra più termini per lo stesso concetto quello preferito. Identifica classi di equivalenza (sinonimi)
- ◆ Es.: regola/norma, week-end/finesettimana, mal di testa/cefalea.

Relazione associativa

- ◆ Relazione residuale, identifica tra due termini una relazione né di equivalenza, né di subordinazione, ma comunque esistente ed innegabile.
- ◆ Es.: barca/nave, ecologia/inquinamento, ecc.

Classificazioni a faccette (1)

Un termine introdotto da S. R. Ranganathan negli anni '30 per indicare la possibilità di descrivere un oggetto complesso attraverso un insieme di affermazioni appartenenti ad uno schema fisso di proprietà, ciascuna delle quali in grado di usare valori da un apposito tesoro.

Ogni risorsa viene descritta dunque dalla tupla di tutti i valori specificati nell'ordine definito dallo schema designato.

Attenzione: lo schema deve anche essere in grado di arrivare ad identificare, e non solo descrivere, una specifica risorsa individuale. Cioè data una tupla intera, debbo trovare zero o una risorsa, non di più.

Questo è solitamente realizzato identificando una (o più) proprietà dette chiave.

Classificazione a faccette (2)

Ogni volta che prevediamo una molteplicità di fattori descrittivi, indipendenti gli uni dagli altri, con cui classifichiamo una risorsa, abbiamo una classificazione a faccette

Ad esempio Dublin Core:

- ◆ Tipo Documento: slide
- ◆ Destinatari: studenti universitari
- ◆ Titolo: Lezione di Semantic Web
- ◆ Autore: Fabio Vitali
- ◆ URL: [http://vitali.web.cs.unibo.it/twiki/ ... /SW.ppt](http://vitali.web.cs.unibo.it/twiki/.../SW.ppt)
- ◆ Formato: MS Powerpoint
- ◆ Data di creazione: 18/2/2011

Classificazione a faccette (3)

```
<HTML>
  <HEAD>
    <META NAME="DC.Title" CONTENT="Il semantic Web">
    <META NAME="DC.Creator" CONTENT="Fabio Vitali">
    <META NAME="DC.Type" CONTENT="text">
    <META NAME="DC.Date" CONTENT="2006-03-27">
    <META NAME="DC.Format" CONTENT="text/html">
    <META NAME="DC.Identifier" CONTENT="http://www...">
  </HEAD>
  <BODY>
    <DIV ID="slide1">
  </DIV>
</BODY>
</HTML>
```

Ontologia

Il culmine della progressione che abbiamo visto finora.

Il principio fondamentale è che il valore di una proprietà non deve necessariamente essere un termine da un vocabolario controllato, ma può essere un riferimento ad un ente, a sua volta descritto da una serie di proprietà.

- ◆ Non è la stringa “Fabio Vitali” ad essere l'autore di queste slide, ma quella persona il cui nome è la stringa “Fabio Vitali”.

Un'ontologia allora è una composizione di classi, in relazione con le altre attraverso il riferimento esplicito (diretto o indiretto) espresso nelle proprietà di uno schema di classificazione a faccette.

Quando i valori di una proprietà sono termini, usiamo un tesaurus, altrimenti relazioni ad altre classi dell'ontologia.

Dai metadati alle ontologie (1/4)

Una collezione di metadati:

- ◆ Fabio Vitali, Bologna University, 18 febbraio 2011, Informatica, Corso di aggiornamento per docenti, Dipartimenti di Filosofia, Università di Bologna, PowerPoint, dispense, ontologie, Metadata and ontologies, raw metadata collections vs. ontologies, using metadata

Un grande casino:

- ◆ Che cos'è un Fabio Vitali?
- ◆ Perché due volte Università di Bologna?
- ◆ Perché alcune parole sono in Italiano e altre in Inglese?
- ◆ Le dispense sono stanze? Cibi? O un tipo di documento?
- ◆ Perché Metadata appare varie volte?

Dai metadati alle ontologie (2/4)

Vocabolario controllato

- ◆ Se restringiamo i metadati a specifici insiemi di termini, riduciamo un po' di ambiguità e di polisemie (dispense) e di varietà (Bologna University, Università di Bologna)

Tassonomie/Thesauri

- ◆ Se organizziamo i termini in un vocabolario controllato possiamo almeno dedurre i significati di qualche concetto
 - ◆ Productivity Tools
 - Computer Applications
 - Microsoft PowerPoint
 - ◆ Documenti testuali
 - Documenti didattici
 - dispense

Dai metadati alle ontologie (3/4)

Classificazione a faccette

- ◆ Associare etichette descrittive a ogni termine ci permette di fare alcune deduzioni forti
 - ◆ Author: Fabio Vitali
 - ◆ Title: Metadata
 - ◆ Subject: Metadata and ontologies, raw metadata collections vs. ontologies, using metadata
 - ◆ Format: Microsoft PowerPoint
 - ◆ Date: 18 febbraio 2011
- ◆ Se poi restringiamo le etichette al set stabilito da un formato (ad esempio Dublin Core) possiamo garantire che le etichette sono usate sempre e sistematicamente da tutti i documenti di una collezione.

Dai metadati alle ontologie (4/4)

Ontologie

- ◆ Se poi specifichiamo che alcuni valori di metadati sono non stringhe, ma riferimenti a concetti complessi, potremmo evitare ulteriori ambiguità e associare metadati e proprietà a quei valori.
- ◆ `Document {`
 - ◆ `Author: person {`
 - `Name: Fabio Vitali,`
 - `Affiliation: organization { name: Università di Bologna } }`
 - ◆ `Title: Semantic Web`
 - ◆ `Coverage: event {`
 - `Contesto: corso {`
 - `Authority: organization {name: Università di Bologna}`
 - `Date: 18 febbraio 2011`
 - `Title: Corso di aggiornamento per docenti }`
 - ◆ `Subject: Metadata and ontologies, raw metadata collections vs. ontologies, using metadata}`

I problemi

Le strutture concettuali viste in precedenza (vocabolario controllato, tassonomia, tesauro, classificazione a faccette, ontologia), indipendentemente dalla loro caratterizzazione:

- ◆ Richiedono personale qualificato per generare e gestire la strutture (vocabolario, concetti e relazioni)
- ◆ Richiedono contemporaneamente competenza di dominio e competenza di classificazione.
- ◆ Può categorizzare solo sui vocaboli e le relazioni previsti.
- ◆ Bisogna che gli utenti siano d'accordo sulla concettualizzazione
- ◆ Ogni concettualizzazione prematura conduce ad un modello incompleto e difficilmente estendibile.
- ◆ Analogamente, è difficile progettare una concettualizzazione in continua evoluzione.
- ◆ Richiedono dunque una progettazione completa e dettagliata prima di iniziare a valutare e descrivere le singole risorse descritte.

Complessivamente, sono un approccio costoso, ingessato, non democratico, centralizzato e riduzionistico. Inoltre scala male su dimensioni veramente grandi (ad esempio il World Wide Web).

Le folksonomie (1)

Un'idea per risolvere questi problemi viene dalle folksonomie (*tassonomie generate dal popolo: folk*).

Attraverso le folksonomie,

- ◆ Gli utenti finali stessi generano (molteplici) termini descrittivi delle risorse
- ◆ Non c'è vocabolario controllato, non c'è modello concettuale
- ◆ Ogni risorsa viene associata ad una categoria totalmente identificata dal termine usato, in proporzione al numero di utenti che usano quel termine per descriverla
- ◆ La prevalenza statistica di alcuni termini su altri rende la risorsa più identificata da quel termine che da altri.
- ◆ Non è possibile fare inferenze o deduzioni sui termini (sono stringhe opache e non ulteriormente analizzabili).

Le folksonomie (2)

Rispetto alle critiche sui modelli visti in precedenza:

- ◆ Il personale qualificato viene sostituito dalla massa degli utenti finali
- ◆ La competenza di dominio è automaticamente presente,
- ◆ Della competenza di classificazione si fa a meno.
- ◆ Non c'è modello concettuale, il vocabolario è totalmente libero.
- ◆ Gli utenti sono automaticamente d'accordo sulla concettualizzazione
- ◆ Il modello (implicit) è sempre incompleto e sempre estendibile (anzi esteso).
- ◆ Non c'è progettazione completa e dettagliata di alcun modello concettuale, né prima né dopo.

Complessivamente, sono un approccio gratuito, flessibile, democratico, decentralizzato e olistico. Inoltre scala benissimo su dimensioni veramente grandi.

Le folksonomie (3)

Ovviamente non sono esenti da critiche

- ◆ Non c'è protezione da ambiguità semantiche
 - ✦ Bush è un ex-presidente o un cespuglio? Apple è una mela o un'azienda di computer? Paris Hilton è una celebrità o un albergo nella capitale francese?
- ◆ Non c'è protezione da variazioni sintattiche, ortografiche, grammaticali, linguistiche, o da termini sinonimici:
 - ✦ Celebrity, celebrities, celebritad, *celerbity*, famous, star, sono termini diversi.
- ◆ Non c'è relazione concettuale tra i termini
 - ✦ Siamese e gatto sono termini diversi.
 - ✦ Cucina e fornelli sono termini diversi
 - ✦ Papa e Benedetto XVI sono termini diversi

Tuttavia, poiché questi limiti sono condivisi da tutti i termini usati, questo corrisponde ad un'annacquatura della precisione della classificazione, e non ad un'incorretta distribuzione dei pesi della classificazione.

Il Semantic Web

TW

- Architettura del Semantic Web
- RDF
- RDF-S
- OWL



Il Web semantico

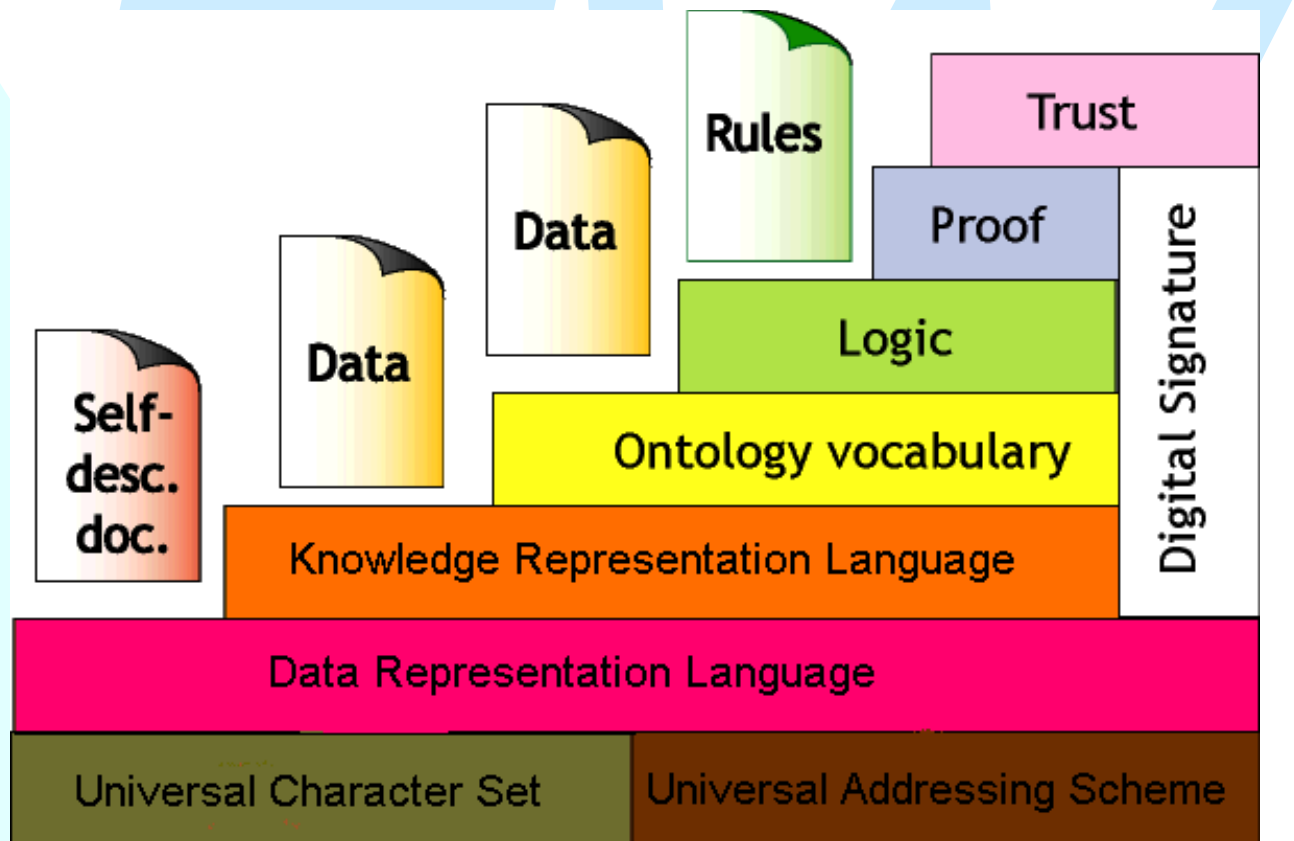
Il W3C considera l'ideale evoluzione del Web dal machine-representable al machine-understandable. L'idea è di generare documenti che possano al tempo stesso essere letti ed apprezzati da esseri umani, ma anche acceduti ed interpretati da agenti automatici alla ricerca di contenuti.

Il Web si deve dunque dotare di una sovrastruttura semantica utilizzabile dalle applicazioni, in modo da poter svolgere quelle funzioni che oggi debbono essere fatte a mano o codificate dentro ai programmi.

Questo porta al web semantico, in cui non esprimo testi (all'interno dei quali le informazioni stanno nascoste e richiedono un umano), ma affermazioni (informazioni non ambigue, che esprimono relazioni tra oggetti, risorse, esseri umani, fatti del mondo reale, e che possono essere utilizzate anche da applicazioni automatiche).

Il Web Semantico (2)

L'idea è di organizzare gli sviluppi futuri del Web Semantico come segue:



XML

Ottimo come sintassi e struttura dati:

- ◆ Elimina ambiguità tra contenuto e markup
- ◆ Elimina incertezze e dipendenze da specifiche codifiche carattere
- ◆ Fornisce API e modelli concettuali semplici per trattare qualunque tipo di struttura dati

Ma non perfettamente adatto per il Semantic Web:

- ◆ Troppi modi "linguistici" per esprimere gli stessi concetti
- ◆ Attributi e entità sono retaggio di un passato di linguaggio per documenti pensati per essere letti.

Meglio trovare un modello astratto per esprimere i concetti, e lasciare ad XML il compito di renderli in maniera linguistica.

RDF

Resource Description Framework è il modello astratto proposto dal W3C per esprimere affermazioni sul mondo.

RDF permette di esprimere ogni affermazione come una tripla (Soggetto, Predicato, Oggetto) (ad es.: "il documento <http://www.cs.unibo.it/~fabio> è stato creato da Fabio Vitali"), dove il soggetto è un URI, il predicato esprime una relazione, e l'oggetto è un'altra risorsa, oppure un valore letterale.

Oltre alle affermazioni, RDF permette di esprimere anche citazioni, ovvero reificazioni, ovvero meta-affermazioni, vale a dire affermazioni su altre affermazioni (es.: "Andrea dice che il documento <http://www.cs.unibo.it/~fabio> è stato creato da Fabio Vitali").

Le triple e RDF

Le triple RDF permettono non solo di associare tra loro dati, ma anche concetti complessi (rappresentati da indirizzi di rete o URI) che permettono di creare discorsi complessi

Dante Alighieri è l'autore della Divina Commedia diventa dunque uno statement che collega il concetto di Dante Alighieri (rappresentato da un URI) al concetto della "Divina Commedia" (anche qui un URI) attraverso il predicato "autore di".

`http://it.wikipedia.org/Dante autore_di http://it.wikipedia.org/Divina_Commedia`

Rinunciare alle stringhe e passare agli URI ci permette di separare i concetti (URI) dai nomi (stringhe) e combinare affermazioni sugli stessi concetti (La Divina Commedia è stata completata nel 1321)

`http://it.wikipedia.org/Divina_Commedia è_completata_nel "1321"`

L'organizzazione di grandi quantità di affermazioni di questo tipo ci permette di fare affermazioni complesse.

Web Ontology Language (1) (OWL)

OWL aggiunge a RDF i concetti di classe e proprietà, permettendo di armonizzare e organizzare le asserzioni fatte in RDF.

OWL permette non più di fare affermazioni su URI rappresentanti concetti singoli (Dante Alighieri), ma come appartenenti a classi di individui descritti e caratterizzati da proprietà.

Ecco allora che Dante è un individuo della classe Persona, che la Divina Commedia è un individuo della classe Poema, e che “autore_di” è una proprietà che lega le persone ai poemi.

Web Ontology Language (2) (OWL)

Per descrivere e organizzare meglio queste classi, è possibile porle in relazione tra loro come generalizzazioni e restrizioni di altre classi.

Ad esempio, è possibile definire la classe Poeta come una sottoclasse di Persona e definire Poema come sottoclasse di Documento.

Quindi, definendo “autore_di” come una relazione tra persone e documenti, le affermazioni

- **<http://it.wikipedia.org/Dante> autore_di http://it.wikipedia.org/Divina_Commedia**
- **http://it.wikipedia.org/Divina_Commedia è un Poema**
ci permette di asserire che
- **<http://it.wikipedia.org/Dante> è una persona**
- **http://it.wikipedia.org/Divina_Commedia è un documento**
- **<http://it.wikipedia.org/Dante> è un poeta**

Abbiamo creato nuova conoscenza applicando regole alle conoscenza esplicite.

Web Ontology Language (1) (OWL)

OWL aggiunge a RDF un ricco vocabolario di descrizione di proprietà e schemi: tra gli altri, relazioni tra classi (es. disgiunzione), uguaglianza, cardinalità (es. “esattamente una”), tipizzazione delle proprietà, caratterizzazione delle proprietà (es. simmetria), ecc.

OWL viene sviluppato in tre diversi sottolinguaggi:

- ◆ OWL Lite
- ◆ OWL DL
- ◆ OWL Full

Web Ontology Language (2)

- OWL Lite: Un modello semplice per esprimere gerarchie classificatorie e semplici vincoli. Permette di esprimere immediatamente tesauri e tassonomie. Ha un modello formale semplice e facilmente trattabile
- OWL DL (Description Logic): alta espressività ma distinzione nei livelli (es.: una classe non può essere istanza di un'altra classe). Garantisce un modello formale trattabile (computabilità in tempi finiti) ma con notevole libertà espressiva
- OWL Full: massima espressività, nessuna garanzia computazionale: le classi possono mescolarsi in gerarchie, è possibile fare affermazioni di ordine superiore. Difficilmente trattabile da applicazioni, è implausibile che applicazioni di reasoning possano supportare ogni caratteristica di OWL Full.

Trust and Proof (1)

Nel momento in cui inizio a realizzare inferenze, ho da considerare anche il problema della veridicità delle informazioni, e della loro affidabilità.

Che deduzioni posso ottenere dalla combinazione di due o più collezioni RDF, se tra di loro esistono affermazioni contraddittorie?

Una fragilità fondamentale dei sistemi di logica del primo ordine è data dal fatto che

- ◆ Affermazioni contraddittorie non generano nuova informazione,
- ◆ Affermazioni contraddittorie possono essere usate per giustificare qualunque inferenza: $A \wedge \neg A \Rightarrow *$.

Se una collezione RDF dice che il cielo è blu, e un'altra dice che il cielo non è blu, io sono logicamente autorizzato a concludere che 4 è dispari o che voi mi dovete €50 a testa.

Trust and Proof (2)

Il passo successivo (ancora tutto da studiare) è creare una rete di affermazioni di affidabilità e fiducia (trust) sulle collezioni, in cui viene espresso il valore di affidabilità delle affermazioni contenute.

Questo passa attraverso:

- ◆ L'ideazione di modelli di contesto all'interno del quale l'affermazione possano essere ritenuta vera. Ad es., "Oggi è sereno" è vera in certi contesti (date, luoghi), e non vera in altri.
- ◆ L'ideazione di modelli di attribuzione (associazione delle affermazioni ai rispettivi autori) che permettano di accettare tutte e sole le affermazioni di gruppi di cui mi fido
- ◆ L'ideazione di sistemi di sicurezza basati su crittografia e firme digitali che permettano di rendere affidabile il sistema di attribuzione

Il *web of trust* è il passo ultimo per permettere la creazione di significato utile, automatico, affidabile su documenti e cose del mondo reale.

Conclusioni

Oggi abbiamo parlato di

- ◆ La necessità del Semantic Web
- ◆ Modelli ontologici per la rappresentazione documentale
- ◆ Modelli per la rappresentazione concettuale

