Strutturare la conoscenza:

XML, RDF, Semantic Web

Oreste Signore

W3C Office in Italy e ISTI-CNR
Area della Ricerca CNR di Pisa – via G. Moruzzi, 1 – 56124 Pisa (Italy)
e.mail: oreste@w3.org
home page: http://www.weblab.isti.cnr.it/people/oreste/

Abstract. Il web è forse oggi il maggior contenitore di conoscenza, o, comunque, è quello più frequentemente utilizzato da una larga varietà di persone. Ma il web, per la sua natura intrinsecamente decentralizzata, rende di importanza vitale l' interoperabilità delle applicazioni a livello tecnologico e semantico. Nell' evoluzione verso il Semantic Web, la strutturazione della conoscenza riveste un ruolo fondamentale. L' aderenza agli standard tecnologici definiti dal W3C (XML, RDF, OWL) consente di realizzare le nuove applicazioni in un contesto aperto e distribuito, coerente con l' evoluzione del web.

Introduzione

La gestione della conoscenza è un tema che va assumendo sempre maggiore importanza, e che funge da catalizzatore per l' integrazione di varie linee di ricerca.
In questo lavoro vengono prima sommariamente illustrate alcune esigenze nel settore del knowledge management, e successivamente descritte, sia pure in modo molto sintetico e semplificato, le caratteristiche del Web e alcune tecnologie W3C che possono fornire una soluzione ai problemi essenziali, consentendo di individuare linee di sviluppo coerenti con il trend evolutivo del Web. In particolare, viene illustrato il ruolo fondamentale giocato da **XML** per *strutturare* l' informazione e supportare l' *interoperabilità all' interno delle applicazioni*, mentre **RDF** consente l' *interoperabilità tra applicazioni*.

Un approccio al Knowledge Management

Due processi essenziali nel Knowledge Management sono da un lato, la possibilità di *reperire le fonti di conoscenza* rilevanti per il problema specifico, e, dall' altro, *fornire le fonti di conoscenza* da utilizzare per risolvere i problemi.
Schematicamente, in un sistema di Knowledge Management possiamo individuare cinque processi:

1. *Acquisizione della conoscenza*: catturando le competenze degli esperti, operando deduzioni da fatti concreti, etc.
2. *Rappresentazione della conoscenza*: memorizzazione delle regole
3. *Elaborazione della conoscenza*: manipolazione della conoscenza memorizzata, per derivare dipendenze o consistenza di regole
4. *Condivisione della conoscenza*: mediante la ricerca di regole che soddisfano la query dell' utente.
5. *Utilizzo della conoscenza*: per risolvere il problema contingente, eventualmente annotando la conoscenza esistente, in modo da renderne disponibile di nuova.

Il Web, e in particolare il Semantic Web, che ne è la naturale evoluzione, costituisce un formidabile componente per supportare gran parte di questi processi. In questo contesto, come verrà dettagliato in seguito, viene utilizzato un formalismo per rappresentare, in forma molto semplice, dei *fatti*, sui quali è possibile operare dei *ragionamenti*. Un elemento significativo è che la conoscenza codificata nel web è rappresentata in maniera *elaborabile dalla macchina*, e quindi può essere utilizzata da componenti automatizzati, denominati *agenti software*.
A puro titolo di esempio, un agente software può comprendere il significato di un' informazione, come, per esempio, che:

*il paziente **X**, affetto dalla patologia **Y**, viene curato con il medicinale **Z***

Il software agent, quindi. può utilizzare questa informazione per mettersi in collegamento con altri agenti software, per esempio per fissare un appuntamento con il dottor **D** (è questo uno degli scenari descritti in [TBL2001]).
Ovviamente, le informazioni utili non sono sempre legate ad un singolo fatto (il particolare paziente, la specifica patologia, una determinata specialità medicinale), ma possono essere relative ad una classe di individui, caratterizzata da alcune proprietà (fascia d' età, sesso, tipo di lavoro, etc.). Quindi, proposizioni atomiche possono essere combinate in maniera più espressiva, per esempio in forma *condizionale*:

Precondizione: "il paziente è di sesso femminile e soffre della patologia **Y**";

Azione: "deve essere curato con il medicinale **Z**".

Dal punto di vista implementativo, tali proposizioni vengono spesso espresse nella forma:

If-Then (**Se-Allora**)

per formare unità elementari per successivi processi di inferenza.
La ricerca di informazioni è uno dei principali punti deboli del web, nonostante il gran numero di motori di ricerca esistenti, che sono poveri di semantica sia in fase di indicizzazione che in fase di ricerca. In fase di indicizzazione, essi utilizzano o moduli compilati dai fornitori di informazioni, che spesso non consentono di specificare metainformazioni come l' autore, le parole chiave, etc., o strumenti automatici (*spider*) per accedere alle pagine ed estrarre semantica. Talvolta, anche le informazioni contenute nei tag `<meta>` vengono di fatto ignorate. In fase di ricerca, viene consentito di combinare le parole con operatori di contesto ("tutte le parole", "una parola qualunque", "nel titolo"), ma in definitiva il risultato scaturisce sempre da una ricerca sulla presenza di parole chiave e dall' identificazione dei documenti più affini alla domanda posta.
L' esistenza di proposizioni più ricche dal punto di vista espressivo permette invece agli utenti di ritrovare in maniera più facile ed efficace le informazioni necessarie per risolvere i problemi. Tra l' altro, la presenza di queste proposizioni condizionali consente di indicizzare le risorse esistenti sul web in maniera più ricca rispetto al metodo tradizionale di associare alle risorse parole chiave o concetti. Diventa allora possibile formulare richieste più sofisticate, migliorando sia la precisione delle risposte ottenute, che il richiamo dei documenti pertinenti. Giusto a titolo di esempio, in assenza di questo tipo di arricchimento, una query che richiedesse la restituzione di tutti i documenti in cui compaiono le parole chiave:

"aspirina" AND "mal di testa"

restituirebbe sia i documenti che descrivono come l' *aspirina **cura** il mal di testa*, sia quelli che descrivono come l' *aspirina **causa** mal di testa*.
Da questa breve descrizione emerge chiaramente l' importanza di disporre di strumenti e tecnologie che permettano di rendere comprensibili a strumenti automatici le proposizioni, e che consentano una reale *interoperabilità tecnologica e semantica*.

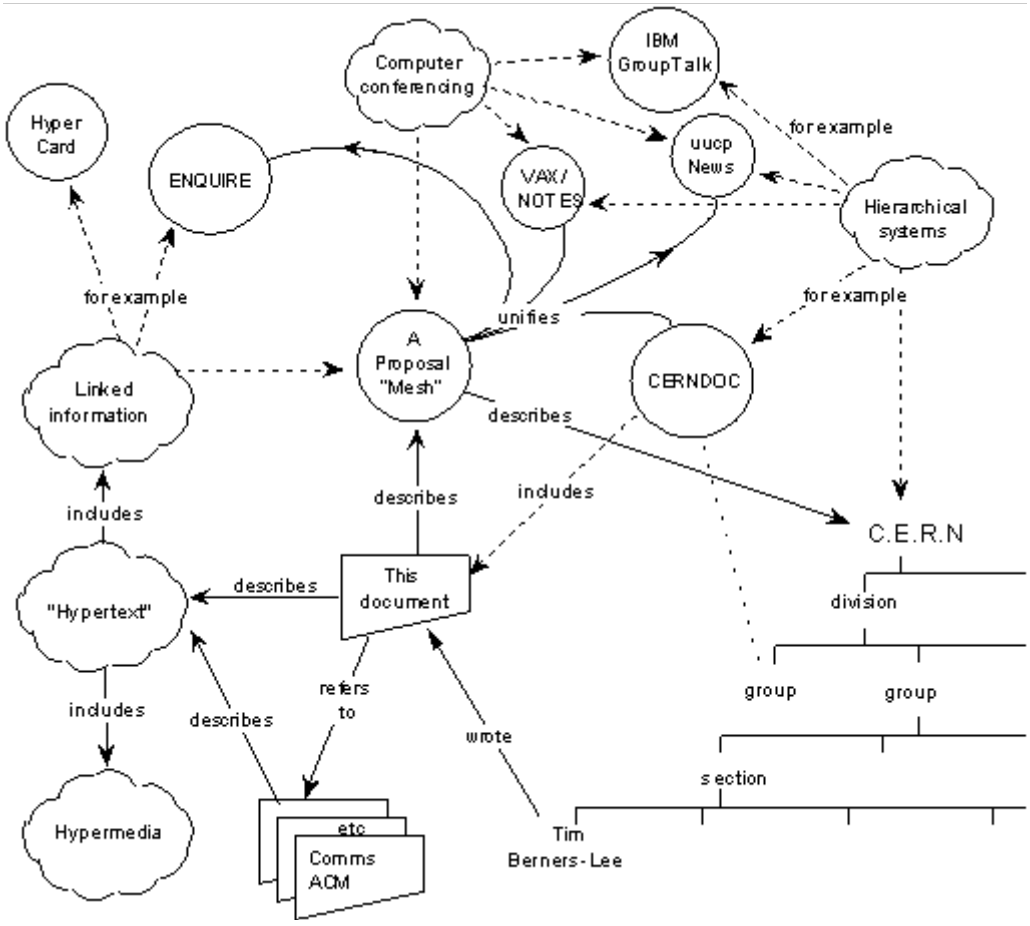

Figura 1 - La visione originaria del World Wide Web

Il Web

Il World Wide Web Consortium (W3C)

Il World Wide Web Consortium (W3C, http://www.w3.org), è un consorzio di imprese che regola l' evoluzione del web, sviluppando tecnologie e definendo protocolli comuni che ne favoriscano l' *evoluzione* e assicurino l' *interoperabilità*. Gli *obiettivi a lungo termine* del W3C possono essere espressi sinteticamente come:

- *Universal Access*: Rendere il Web accessibile a tutti, promuovendo tecnologie che tengono conto delle notevoli differenze in termini di cultura, formazione, capacità, risorse materiali, e limitazioni fisiche degli utenti in tutti i continenti.
- *Semantic Web*: Sviluppare un ambiente software che consenta ad ogni utente di fare il miglior uso possibile delle risorse disponibili sul Web.

- *Web of Trust*: guidare lo sviluppo del Web tenendo in attenta considerazione gli aspetti innovativi che questa tecnologia solleva in campo legale, commerciale e sociale.

Le tecnologie W3C costituiscono un insieme di strumenti che permettono di strutturare, condividere e utilizzare, la conoscenza, in un ambiente aperto e in costante evoluzione.

I principi informatori

Il Web è una applicazione costruita su Internet, e quindi ne ha ereditato i principi fondamentali, che, molto sinteticamente, sono:

- *Interoperabilità*: Le specifiche dei linguaggi e dei protocolli del Web devono essere compatibili tra di loro, e consentire a qualunque hardware e software di operare tra di loro.
- *Evoluzione*: Il Web deve essere in grado di accogliere le nuove tecnologie. Principi di progettazione quali la semplicità, la modularità e l' estensibilità aumentano le possibilità che il Web sia in grado di funzionare con le tecnologie emergenti, quali i dispositivi mobili e la televisione digitale, o con altre tecnologie che compariranno.
- *Decentralizzazione*: La decentralizzazione è senza dubbio il principio più nuovo e difficile da applicare. Per consentire che il Web si diffonda realmente su scala mondiale senza rischiare errori o interruzioni, l' architettura (e Internet) devono limitare o eliminare le dipendenze da nodi centrali.

L' interoperabilità

Due applicazioni sono interoperabili se si possono scambiare dati e servizi in modo efficace e consistente, permettendo la comunicazione tra piattaforme hardware e software eterogenee. Nel contesto attuale, l' interoperabilità costituisce un fattore chiave di successo. I vantaggi sono ben noti, e comprendono la possibilità di salvaguardare gli investimenti fatti, grazie alla possibilità di adeguarsi all' evoluzione degli ambienti operativi, e quella di allargare il numero di applicazioni con cui interagire. La coerenza con un contesto tecnologico solido è un elemento fondamentale per raggiungere un buon livello di interoperabilità. Tuttavia, l' interoperabilità non è un aspetto meramente tecnologico. Bisogna tener presenti le differenti culture e il diverso modo di percepire i concetti, quindi occorre considerare non solo l' *interoperabilità tecnologica*, ma anche quella *semantica*.

L' obiettivo dell' Universal Access, in particolare, pone l' enfasi sul superamento delle differenze di cultura, lingua, formazione, capacità, risorse materiali, limitazioni fisiche e cognitive degli utenti in tutti i continenti. Quindi, in un web che voglia essere davvero universale, vanno considerate, e superate, anche le potenziali *barriere culturali* determinate dalle differenze di tradizione e e di storia degli utenti.

Espressioni, colori, immagini, classificazioni di concetti, possono essere totalmente diversi per persone di culture diverse. Anche l' aspetto esteriore può determinare un diverso modo di percepire il messaggio, che è costituito da:

- *Contenuto*: il contenuto reale del messaggio, che l' autore intende comunicare;
- *Struttura*: il modo in cui è organizzata l' informazione (es. titolo, autore, corpo del testo, firma);
- *Presentazione*: il modo in cui l' informazione viene presentata all' utente (tipo di carattere, colore, organizzazione della pagina, etc.).

Ciascuno di questi componenti ha una *valenza semantica*, e veicola una *conoscenza esplicita o tacita*. Condividere la conoscenza sul web significa poter disporre di strumenti e tecnologie che consentano di *esprimere i contenuti, strutturarli e presentarli* in modo adeguato, rendendone esplicita la *semantica* e consentendo la fruizione dell' informazione a tutti, indipendentemente dal particolare retroterra culturale e dal contesto tecnologico.

.

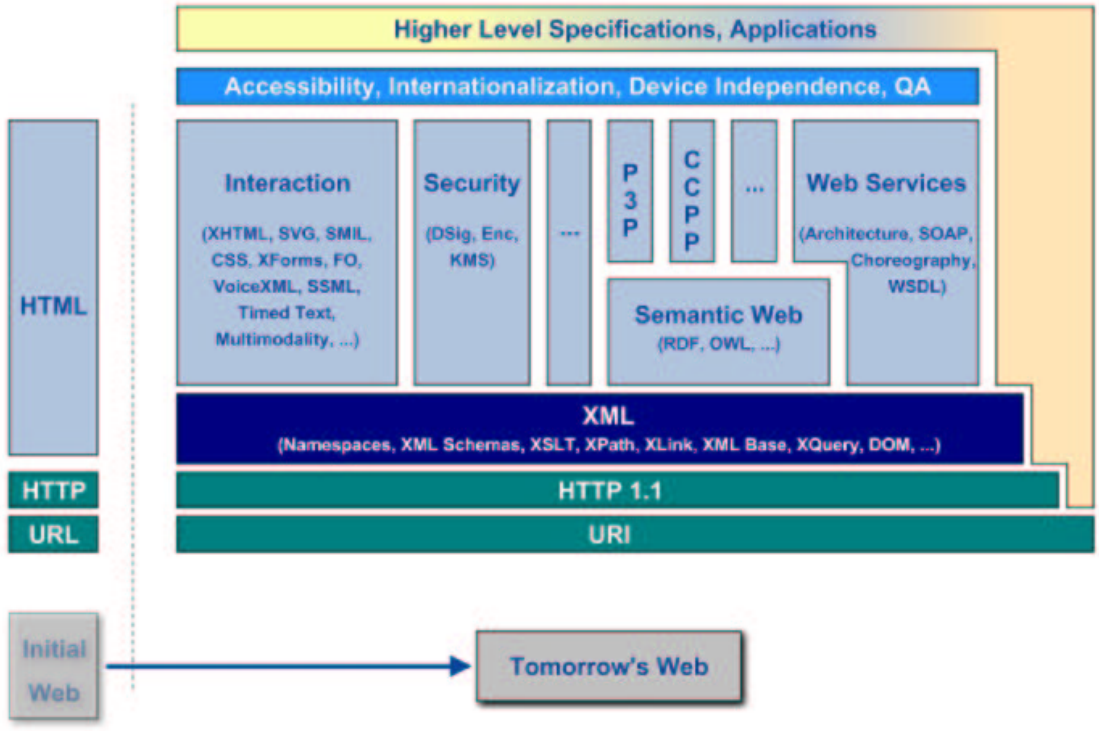

Figura 2 - L'evoluzione del Web

L'evoluzione

Il World Wide Web, nato da un' idea di Tim Berners Lee (Figura 1), ha avuto un impatto incredibile sulla nostra realtà, modificando radicalmente il modo in cui le persone accedono all' informazione, e cambiando la vita di milioni di persone. Nel giro di pochi anni, vi è stata una notevole evoluzione del web (Figura 2), con uno spostamento dell' enfasi dall' *interazione uomo-macchina* all' *interazione macchina-macchina*. Si noti, in Figura 1, come gli archi rechino una esplicita identificazione del loro significato (*semantica dei link*).

Decentralizzazione e architettura peer-to-peer

Internet, progettata originariamente come un sistema *peer-to-peer*, si è successivamente trasformata in un ambiente sempre più client-server, in cui milioni di utenti comunicano con un insieme relativamente limitato di server privilegiati. Le applicazioni peer-to-peer attualmente presenti in Internet utilizzano la rete essenzialmente per come era stata concepita: un *mezzo di comunicazione tra macchine che condividono risorse in modo paritetico*. I sistemi peer-to-peer sono del tutto coerenti con una architettura

decentralizzata. Nei sistemi completamente decentralizzati non solo ogni host ha la stessa dignità, ma non esistono host con specifici ruoli di gestione o di supporto.
La realizzazione di sistemi completamente peer-to-peer può risultare complessa, e viene spesso adottata una architettura ibrida (un esempio tipico è la gestione del DNS, basata su un protocrollio peer-to-peer, ma con una struttura gerarchica intrinseca). In pratica, poi, alcune applicazioni operano meglio in un' architettura centralizzata, senza far ricorso a nessuna tecnologia peer-to-peer. La ricerca su grossi database, relativamente statici, è un caso in cui l' adozione di tecnologie e architetture peer-to-peer non reca vantaggi.
La definizione centralizzata di concetti, dizionari, thesauri, e l' identificazione di siti privilegiati, depositari di questa conoscenza, è, invece, un processo che comporta spesso tempi lunghi, anche per superare differenze culturali o tradizioni ben consolidate. Vedremo in seguito come un uso coerente delle tecnologie web consenta di superare questi ostacoli..

L' interoperabilità tecnologica

Strutturazione dell' informazione

XML

Extensible Markup Language (**XML**) è nato per far fronte alle limitazioni di HTML nella realizzazione delle nuove applicazioni Web, in cui i dati costituisconoo un elemento essenziale (*data-centric Web applications*). XML è stato quindi il primo passo per assegnare una semantica ai tag e supportare le transazioni sul Web, permettendo lo scambio di informazioni tra database diversi. Ulteriori e significativi vantaggi sono costituiti dalla possibilità di avere viste diverse degli stessi dati, e la possibilità di personalizzare le informazioni mediante opportuni agenti. L' adozione di XML agevola la gestione di collezioni di documenti, e costituisce un supporto fondamentale per la pubblicazione di informazioni a livello internazionale, con il non piccolo vantaggio di essere indipendente dalla piattaforma e dal linguaggio.
Non a caso XML è stato definito "ASCII del 2000".
Le caratteristiche di XML possono essere illustrate con un esempio semplice, relativo alla gestione degli Ordini.(Figura 3). La sintassi XML usa tag di inizio e fine, come per esempio `<importo>` e `</importo>`, per marcare i campi informativi. Un campo informativo racchiuso tra due marcatori viene detto elemento (*element*) e può essere ulteriormente arricchito dalla presenza di coppie nome/valore (nell' esempio, id="ord001") dette attributi (*attribute*). Come si può vedere, si tratta di una sintassi semplice, la cui elaborazione automatica è poco complessa, senza codifiche particolarmente criptiche, per cui resta comprensibile alla lettura diretta. I tag devono essere inseriti correttamente uno dentro l' altro, deve esistere una corrispondenza tra il tag di apertura e quello di chiusura, sono previsti elementi a campo informativo nullo e gli attributi dei tag devono essere racchiusi tra doppi apici.
La presenza di una struttura formale del documento, espressa nella **DTD** *(Document Type Definition*), non ha un impatto diretto sul modello strutturale implicito: nell' esempio di Figura 3, in cui la DTD è inclusa nel documento (ma potrebbe anche essere referenziata come risorsa esterna) la riga 6 specifica che l' attributo *db* è

obbligatorio.Un documento XML si dice "*well formed*" quando rispetta le regole di scrittura; viene detto "*validato*" quando è coerente con la struttura definita nella DTD.

```
01  <?xml version="1.0"?>
02  <!DOCTYPE ordine [
03  <!ELEMENT ordine ( cliente, prodotto+ )>
04  <!ATTLIST ordine id ID #REQUIRED>
05  <!ELEMENT cliente EMPTY>
06  <!ATTLIST cliente db CDATA #REQUIRED>
07  <!ELEMENT prodotto  ( importo )>
08  <!ATTLIST prodotto  db CDATA #REQUIRED>
09  <!ELEMENT importo   ( #PCDATA )>
10  ]>
11  <ordine id="ord001">
12    <cliente db="codcli123"/>
13    <prodotto db="prod345">
14      <importo>23.45</importo>
15    </prodotto>
16  </ordine>
```

Figura 3 - Un documento XML (con la sua DTD in grassetto)

XML, mezzo espandibile e flessibile per modellare il Web, costituisce attualmente la tecnologia chiave di W3C. XML ricopre un ruolo centrale nell' architettura del Web, e ogni nuovo linguaggio utilizzato per definire un nuovo standard deve essere descritto in XML. W3C considera XML come una famiglia di tecnologie, e non intende centralizzarne il controllo, preferendo lasciare agli utenti, coerentemente con la filosofia del Web, il compito di sviluppare applicazioni particolari.

XML Schema Definition

La DTD presenta alcune limitazioni, riconducibili essenzialmente al fatto che viene espressa con una sintassi sua propria, e quindi richiede editor, parser e processor ad hoc. Inoltre, è difficile estenderla, non contempla datatype e deve supportare tutti gli elementi e attributi descritti dai namespace[1] inclusi.
Gli schema hanno le stesse funzionalità delle DTD, ma offrono alcuni significativi vantaggi: sono espressi con la sintassi XML e includono datatype, inheritance, regole di combinazione degli schema. **XMLSchema** fornisce anche un miglior supporto dei namespace e offre la possibilità di agganciare documentazione e informazioni semantiche. XMLSchema permette di rappresentare vincoli sui possibili valori, tipi complessi e gerarchie di tipi. In definitiva, gli XMLSchema sono strumenti molto più potenti delle DTD, e sul sito W3C sono disponibili parser, validatori, e altri strumenti utili. Utilizzare questa specifica nella realizzazione di nuove applicazioni costituisce un indubbio investimento per il futuro.

Presentazione dell' informazione

L' architettura di riferimento

Le applicazioni XML sono ormai moltissime, e molte di esse si basano sulla comprensione di un principio chiave: la *separazione tra contenuto e forma*. Ne scaturisce una architettura comune di riferimento (Figura 4), nella quale le informazioni,

[1] Un **XML namespace** è un insieme di nomi, caratterizzati da un URI di riferimento, utilizzati come element type e attribute name. Il concetto sarà chiarito successivamente da un esempio.

estratte dalla base dati aziendale, vengono strutturate in un documento XML, che viene successivamente trasformato nel formato più adatto per l' utente finale, mediante una trasformazione di stile. Si noti che la presentazione dell' informazione non riveste un ruolo semplicemente formale ed estetico, ma investe aspetti di distribuzione dell' informazione (per es. consentendo l' accesso all' informazione in maniera indipendente dal dispositivo utilizzato) e semantici. In particolare, non si può ignorare che la forma in cui viene presentata l' informazione porta intrinsecamente un messaggio preciso, legato alle specifiche tradizioni culturali.

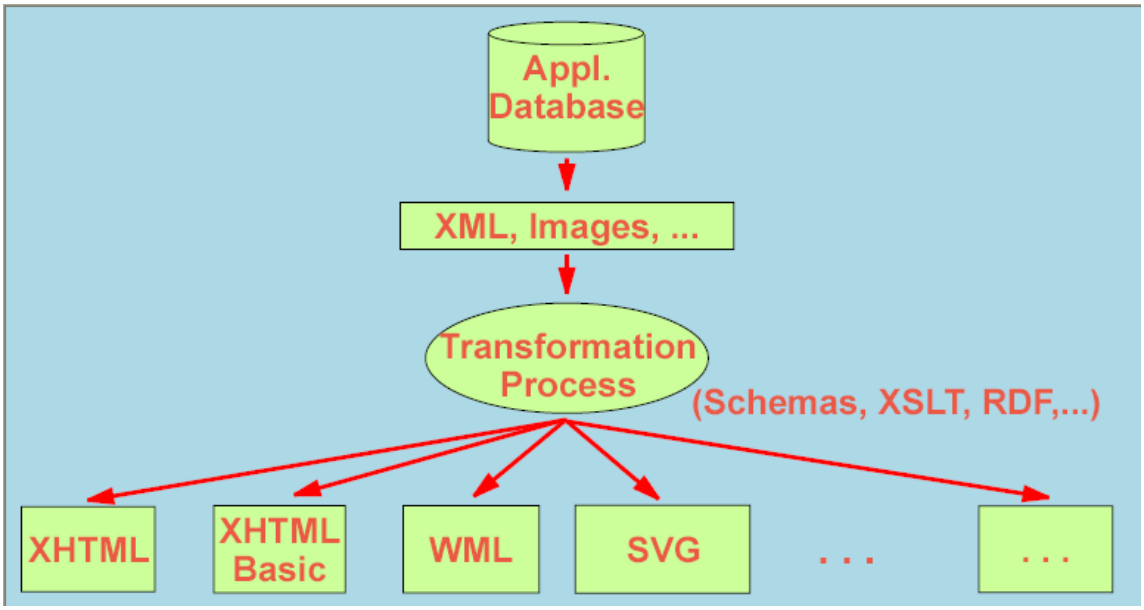

Figura 4 - L' architettura di riferimento

Multimedialità, matematica, grafica

L' arricchimento dell' informazione presentata all' utente è anche legata alla possibilità di corredarla di componenti multimediali o grafiche (si pensi, banalmente, ad un thesaurus corredato di informazioni grafiche, al video che illustra un certo tipo di processo, etc.).

Synchronized **M**ultimedia **I**ntegration **L**anguage (SMIL, pronunciato come la parola inglese "smile") permette in modo semplice la creazione ("authoring") di presentazioni interattive audiovisive. SMIL viene utilizzato tipicamente per presentazioni "rich media"/multimedia, che integrano streaming audio e video con immagini, testi, o altri tipi di media. In termini molto semplici, permette agli autori di specificare cosa deve essere presentato e quando, consentendo agli autori di controllare esattamente quando deve essere pronunciata una certa frase, facendola eventualmente coincidere con la presentazione di una certa immagine sullo schermo. L' idea di base è quella di dare un nome, con una URI[2], ai vari componenti, qualunque sia il medium utilizzato (testo, imagine, video, audio) e programmare la loro presentazione in sequenza o in parallelo. La presentazione è costituita da vari componenti di tipo diverso (audio, video, imagine,

[2] **URI** (*Uniform Resource Identifier*). è il generico insieme di tutti i nomi/indirizzi che costituiscono le brevi sequenze di caratteri che fanno riferimento ad una risorsa.
URL (*Uniform Resource Locator*) è un termine informale, non più utilizzato nelle specifiche tecniche, associato con gli schemi URI più noti e diffusi (http, ftp, mailto, etc.).
Per ulteriori dettagli, si veda [Naming])

testo), memorizzati su unWeb server, quindi accessibili mediante una URI, e l' utente può seguire eventuali hyperlink inclusi nella presentazione.
Sono ovviamente disponibili tutte le funzioni tipiche degli ambienti multimediali. Il linguaggio SMIL è stato progettato in modo che una presentazione possa essere preparata utilizzando un banale editore di testi, raccogliendo così l'eredità di HTML, che permette di creare pagine ipertestuali senza far ricorso a strumenti sofisticati. Esempi classici di applicazioni SMIL sono: archivi di foto digitalizzate, coordinate con una loro presentazione, corsi di formazione con integrazione di voce e immagini, etc.
Molta conoscenza è anche racchiusa, e difficilmente reperibile, in formule matematiche o grafici.

```
<mrow>
  <mi>x</mi> <mo>=</mo>
  <mfrac>
    <mrow>
      <mrow><mo>-</mo><mi>b</mi></mrow>
      <mo>&PlusMinus;</mo>
      <msqrt>
        <mrow>
          <msup><mi>b</mi><mn>2</mn></msup>
          <mo>-</mo>
          <mrow>
            <mn>4</mn><mo>&InvisibleTimes;</mo>
            <mi>a</mi><mo>&InvisibleTimes;</mo>
            <mi>c</mi>
          </mrow>
        </mrow>
      </msqrt>
    </mrow>
    <mrow>
      <mn>2</mn><mo>&InvisibleTimes;</mo>
      <mi>a</mi>
    </mrow>
  </mfrac>
</mrow>
```

Figura 5 - Una formula espressa inMathML

Formule semplici possono essere facilmente inserite nella pagine HTML, utilizzando apici e pedici (per esempio, a_i verrebbe espresso come: `a<sub>i</sub>`). Per formule più complesse (anche la banale formula risolutiva dell' equazione di secondo grado) bisogna però ricorrere ad artifici: tipicamente, rappresentare la formula come immagine, e inserirla poi nel testo. Evidenti, e noti, i problemi di posizionamento delle formule. Inoltre, la formula viene vista come un elemento singolo, nessuna sua parte è individuabile separatamente, la sua presentazione ad un portatore di handicap visivo potrebbe essere problematica. Infine, non è trascurabile il costo di una eventuale trasposizione del materiale esistente in un nuovo formato utilizzabile sul Web.
MathML ([MathML]) è un linguaggio di marcatura che permette di scrivere formule matematiche anche molto complesse (la Figura 5 è un esempio di scrittura di una formula semplice). Le formule sono espresse in XML, e quindi è possibile ricercarne i singoli elementi, si possono mescolare le formule con altri markup, un voice browser potrebbe essere in grado di leggere la formula. In altri termini, le *informazioni semantiche* presenti nella formula possono essere rese esplicite e condivise con altri.
Per la grafica vettoriale esiste una specifica, SVG (**S**calable **V**ector **G**raphics). SVG è un linguaggio per descrivere grafici bidimensionali in XML. SVG gestisce tre tipi di oggetti grafici: forme in grafica vettoriale (per es. cammini, o path, costituiti da linee

rette e curve), immagini e testi. Gli oggetti grafici SVG hanno la proprietà di essere scalabili, con componenti identificabili singolarmente, che possono essere corredati di *descrizioni semantiche* (*metadati*), ed essere origine o destinazione di link. Il linguaggio SVG è molto ricco, e consente anche animazioni.

Design for all: l' accessibilità dei siti Web

Il Web è la tecnologia che si è diffusa più rapidamente nella storia dell' uomo, e sta diventando una risorsa chiave per il *reperimento dell' informazione* (notizie, commercio, entertainment), la *formazione* (classroom education, distance learning), *lavoro* (ricerca d' impiego, interazione sul posto di lavoro), *partecipazione civica* (leggi, elezioni, informazioni governative, servizi). Eppure il Web è talvolta **non accessibile** ai disabili. Ma l' accesso universale è, secondo Tim Berners-Lee, Direttore del W3C e inventore del World Wide Web uno dei requisiti essenziali del web ("*The power of the Web is in its universality. Access by everyone regardless of disability is an essential aspect.*.").

La *Web accessibility* deve considerare vari tipi di disabilità, poiché il Web può presentare ostacoli a persone che abbiano limitazioni visive, uditive, fisiche, cognitive o neurologiche. Anche se non tutte le disabilità hanno impatto sulle possibilità di accesso al Web, va tenuto presente che talvolta anche l' avanzamento nell' età può comportare una combinazione di problemi (diminuzione della vista o dell' udito, riduzione della destrezza, difficoltà di memoria). La *Web accessibility* è importante anche perché milioni di persone hanno difficoltà nell' accesso al Web, e molti Governi, tra cui quello Italiano, hanno emanato linee guida per garantire l' accessibilità dei siti. La Web accessibility ha una valenza non solo *sociale*, ma anche *economic*a (costituisce un mercato rilevante, dato l' elevato numero di portatori di handicap e l' aumento dell' età media) e *tecnologica*, dato che la progettazione che tiene conto dei potenziali handicap porta dei benefici a tutti gli utenti, quando si trovano in condizioni ambientali difficili (dispositivi mobili, eccessiva illuminazione, elevato rumore di fondo, banda limitata, mani e occhi impegnati). Quindi, l' accessibilità contribuisce ad una migliore progettazione per tutti gli utenti, coerentemente con uno dei principi fondamentali del Web: l' Universal Access.

La *Web Accessibility Initiative* (**WAI**) del W3C ha operato in modo efficace per assicurare che le tecnologie Web supportino l' accessibilità, e ha sviluppato alcune Guideline che giocano un ruolo critico nel rendere accessibile il Web: ([WCAG], [ATAG], ([UAAG])..

L' interoperabilità semantica

I metadati

Nel navigare sul web, si seguono dei link, che portano a quella che formalmente viene detta *risorsa* (resource) identificata univocamente da un URI. Nel linguaggio corrente una risorsa viene anche detta "documento", per mettere in evidenza il fatto che sia leggibile da un essere umano, o "oggetto", per mettere in evidenza che è leggibile da una macchina. Qualunque sia il termine utilizzato, la risorsa non è una entità a sé, ma è accompagnata da informazioni che la descrivono. Le informazioni sulla risorsa vengono generalmente dette Metadati.

Si può quindi dire che *i metadati sono informazioni, comprensibili dalla macchina, relative a una risorsa web o a qualche altra cosa*. Il punto chiave è costituito appunto dal fatto che i metadati sono comprensibili dalla macchina (*machine understandable*). Di conseguenza, i metadati costituiscono un tipo di informazione che può essere utilizzata dai *software agent*, per fare un uso appropriato delle risorse, rendendo più semplice e veloce il funzionamento del Web, aumentando la nostra fiducia in esso. A titolo di esempio, quando si reperisce un documento (o un oggetto) sul web, utilizzando il protocollo HTTP, è possibile che il server invii alcune informazioni sulla risorsa, quali la sua data di aggiornamento, la data massima di validità dell' informazione, il suo autore, etc. Quindi il Web, come insieme di risorse e di informazioni sulle risorse (cioè metadati) è già una realtà alla quale siamo abituati.
Va tenuto presente che *i metadati sono dati*, e questo fatto ha alcune conseguenze:

- possono essere *memorizzati come dati*, in una risorsa, che può quindi contenere informazioni relative a se stessa o ad un' altra risorsa. I metadati relativi ad un documento possono essere contenuti nel documento, oppure contenuti in un documento separato, oppure essere trasferiti a corredo del documento.
- possono essere *descritti da altri metadati*, e così via.

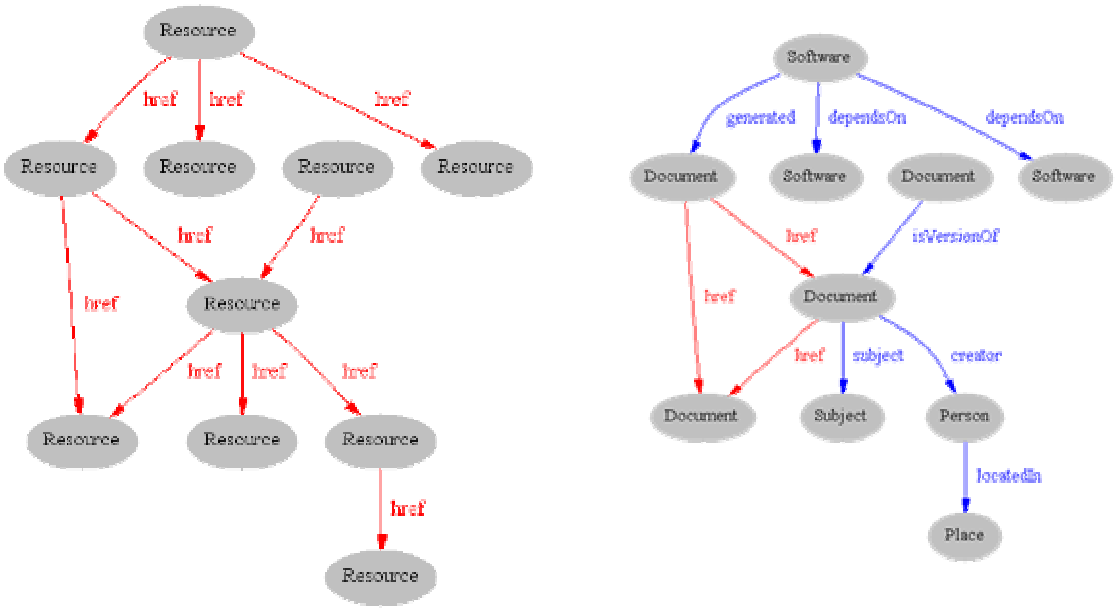

Figura 6 - Il web prima e dopo RDF

Il Resource Description Framework

Automatizzare il Web restando ancorati alla sua architettura originaria, in cui tutte le informazioni erano *machine-readable*, ma non *machine-understandable*, era un obiettivo molto difficilmente raggiungibile, mentre i *metadati* sembrano offrire una soluzione al problema. L' uso efficace dei metadati, tuttavia, richiede che vengano stabilite delle convenzioni per la *semantica*, la *sintassi* e la *struttura*. Le singole comunità interessate alla descrizione delle loro risorse specifiche definiscono la semantica dei metadati pertinenti alle loro esigenze. La sintassi, cioè l' organizzazione sistematica dei data element per l' elaborazione automatica, facilita lo scambio e l'

utilizzo dei metadati tra applicazioni diverse. La struttura può essere vista come un vincolo formale sulla sintassi, per una rappresentazione consistente della semantica.
Resource Description Framework (**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. RDF consente l' elaborazione automatica delle risorse reperibili sul web, e può essere utilizzato e portare vantaggi sono in molti settori, quali, ad esempio:

- *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..

RDF 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** ([RDFMSS]), che definisce il *data model* RDF e la sua codifica XML, e **RDF Schema** ([RDFSS]) che permette di definire specifici *vocabolari* per i metadati.
La Figura 6 mostra l' impatto di RDF sul web.

RDF Data Model

RDF fornisce un modello per descrivere le risorse, che hanno delle proprietà (o anche attributi o caratteristiche). RDF definisce una risorsa come un qualsiasi oggetto che sia identificabile univocamente mediante un Uniform Resource Identifier (URI).
Il data model RDF, che consente di rappresentare statement RDF in modo sintatticamente neutro, è molto semplice ed è basato su tre tipi di oggetti:

Resources Qualunque cosa descritta da una espressione RDF viene detta risorsa (*resource*). Una risorsa può essere una pagina Web, o una sua parte, o un elemento XML all' interno del documento sorgente. Una risorsa può anche essere un' intera collezione di pagine web, o anche un oggetto non direttamente accessibile via Web (per es. un libro, un dipinto, etc.). Le risorse sono sempre individuate da un URI, eventualmente con un anchor id. Qualunque cosa può essere identificata da un URI.

Properties Una *property* (proprietà) è un aspetto specifico, una caratteristica, un attributo, o una relazione utilizzata per descrivere una risorsa. Ogni proprietà ha un significato specifico, definisce i valori ammissibili, i tipi di risorse che può descrivere, e le sue relazioni con altre proprietà. Le proprietà associate alle risorse sono identificate da un *nome*, e assumono dei *valori*.

Statements Una risorsa, con una proprietà distinta da un nome, e un valore della proprietà per la specifica risorsa, costituisce un RDF *statement*. Uno statement è quindi una tupla composta da un *soggetto* (risorsa), un *predicato*

(proprietà) e un *oggetto* (valore). L' oggetto di uno statement (cioè il property value) può essere un' espressione (sequenza di caratteri o qualche altrro tipo primitivo definito da XML) oppure un' altra risorsa.

Graficamente, le relazioni tra Resource, Property e Value vengono rappresentate mediante *grafi etichettati orientati*, in cui le risorse vengono identificate come nodi (graficamente delle ellissi), le proprietà come archi orientati etichettati, e i valori corrispondenti a sequenze di caratteri come rettangoli. Un insieme di proprietà che fanno riferimento alla stessa risorsa viene detto descrizione (*description*).
RDF permette di descrivere anche fatti complessi. Per esempio, il fatto espresso dalla concatenazione delle due frasi:

> *La persona identificata dal Codice Fiscale SGNRST99A99X111Y ha Name Oreste Signore, Email oreste@w3.org, e Affiliation C.N.R..*
> *La risorsa http://www.w3c.it/Oreste/DocX ha come Author questa persona*

verrebbe rappresentato dal diagramma di Figura 7

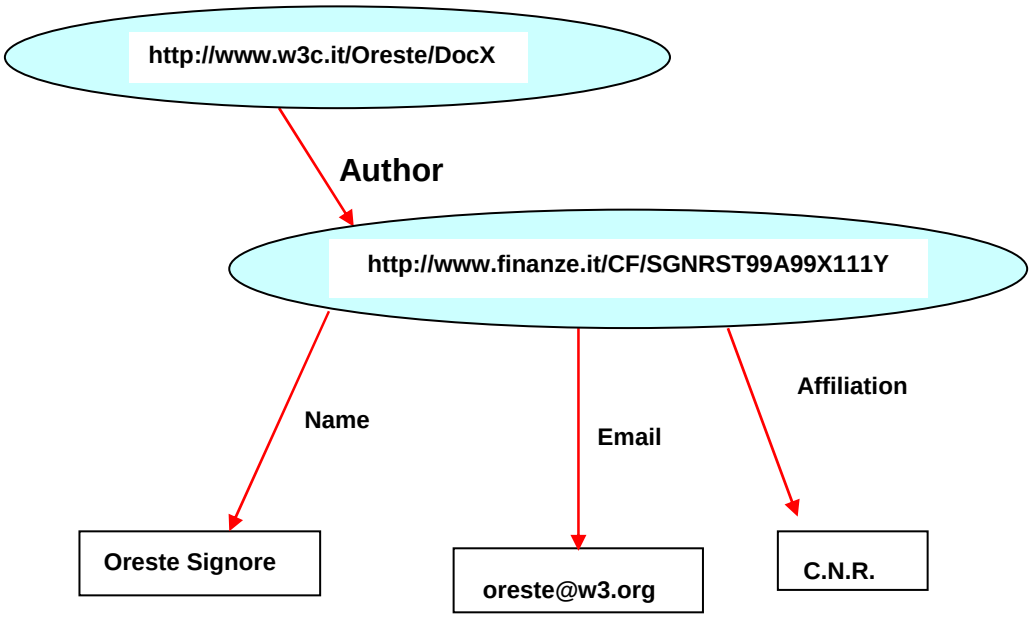

Figura 7 – La rappresentazione grafica di due statement RDF concatenati

In questo esempio è stata creata una risorsa, identificabile univocamente, per l' autore, ma non per il nome, la Email, l' affiliazione. Il modello RDF consente la creazione di risorse a più livelli. Per esempio, sarebbe stato possibile creare una risorsa per l' Affiliazione, con proprietà come: tipoDiEnte, partita IVA, sedeSociale, etc. I limiti pratici e logici per il numero di livelli dipendono essenzialmente dalle caratteristiche e dalle tradizioni delle singole comunità che definiscono la descrizione delle risorse.

Namespace: un elemento chiave nell' architettura peer-to-peer

RDF supporta l' utilizzo di convenzioni che rendono più agevole l' *interoperabilità tra insiemi separati di metadati*. Queste convenzioni includono l' utilizzo di meccanismi standard per rappresentare la semantica, basati sul modello, semplice ma potente, illustrato precedentemente. Inoltre, RDF consente di pubblicare vocabolari *machine readable*, ma anche leggibili da utenti umani. I vocabolari sono un insieme di proprietà

(o *metadata elements*) definiti dalle singole comunità disciplinari. La capacità di standardizzare le definizioni dei vocabolari potrebbe favorire enormemente il riuso e l' estensione della semantica tra comunità diverse.
RDF identifica univocamente le proprietà mediante il meccanismo dei namespace XML ([XMLns]), che forniscono un metodo per identificare in maniera non ambigua la semantica e le convenzioni che regolano l' utilizzo delle proprietà identificando l' authority che gestisce il vocabolario.
Uno degli esempi più noti è la Dublin Core Initiative ([DC]) che definisce, per esempio, il campo "*Subject and Keywords"* nel seguente modo:

```
Name:           Subject and Keywords
  Identifier:   Subject
  Definition:   The topic of the content of the resource.
  Comment:      Typically, a Subject will be expressed as keywords, key
                phrases or classification codes that describe a topic
                of the resource.
                Recommended best practice is to select a value from a
                controlled vocabulary or formal classification scheme.
```

Si può utilizzare un namespace XML per identificare in maniera non ambigua lo schema per il vocabolario Dublin Core puntando alla risorsa Dublin Core che ne definisce la semantica. La descrizione di un sito Web mediante le proprietà definite nel vocabolario Dublin Core e quelle di una personale estensione potrebbe essere:

```
<rdf:RDF
  xmlns:rdf=http://www.w3.org/1999/02/22-rdf-syntax-ns#
  xmlns:dc="http://purl.org/metadata/dublin_core#"
  xmlns:mydc="http://www.w3c.it/metadata/DCaddendum#">
  <rdf:Description about="http://www.dlib.org">
    <dc:Title>
        D-Lib Program - Research in Digital Libraries
    </dc:Title>
    <dc:Description>
    The D-Lib program supports the community of people with research
          interests in digital libraries and electronic publishing
    </dc:Description>
    <dc:Publisher>
        Corporation For National Research Initiatives
    </dc:Publisher>
    <dc:Date>1995-01-07</dc:Date>
    <dc:Subject>
      <rdf:Bag>
        <rdf:li>Research; statistical methods</rdf:li>
        <rdf:li>Education, research, related topics</rdf:li>
        <rdf:li>Library use Studies</rdf:li>
      </rdf:Bag>
    </dc:Subject>
    <dc:Type>World Wide Web Home Page</dc:Type>
    <dc:Format>text/html</dc:Format>
    <dc:Language>en</dc:Language>
    <mydc:Rating>
       Well known and often referenced site
    </mydc:Rating>
    <mydc:Originality>High</mydc:Originality>
  </rdf:Description>
</rdf:RDF>
```

Si noti in questo esempio, che costituisce una variante di uno di quelli presentati in [RDFMSS], la presenza di tre *namespace*, referenziati dai prefissi **rdf**, **dc** e **mydc** che permettono di utilizzare le proprietà definite nei tre namespace. In particolare, il namespace mydc permette di ampliare il numero di proprietà definite dal namespace che referenzia Dublin Core.

Il Semantic Web

La sfida dei prossimi anni è il Semantic Web, che, nella visione di Berners-Lee, ha una architettura a livelli (Figura 8).

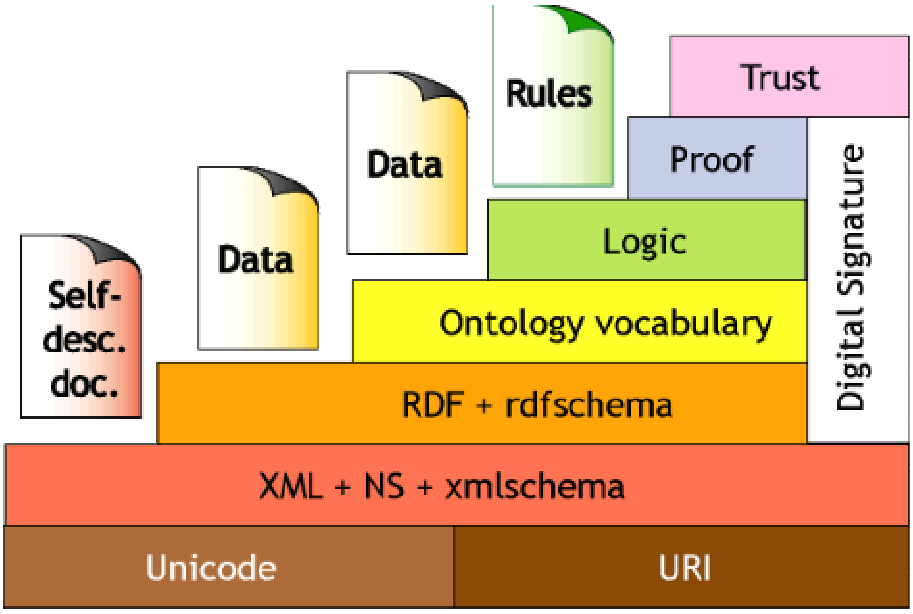

Figura 8 - L' architettura del Semantic Web
(http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html)

Per chiarezza di terminologia, va ricordato che la filosofia di base del Web è quella di uno spazio informativo universale, navigabile, con un mapping da *URI (Uniform Resource Identifier*) alle risorse. Nel contesto del Semantic Web, il termine semantico assume la valenza di *"elaborabile dalla macchina*" e non intende fare riferimento alla semantica del linguaggio naturale e alle tecniche di intelligenza artificiale. Il Semantic Web è, come l' XML, un ambiente dichiarativo, in cui si specifica il significato dei dati, e non il modo in cui si intende utilizzarli. La semantica dei dati consiste nelle informazioni utili perché la macchina possa utilizzarli nel modo corretto, eventualmente convertendoli.
Come chiaramente descritto in [TBL2001], il Semantic Web potrà funzionare solo se le macchine potranno accedere ad un *insieme strutturato di informazioni* e a un insieme di *regole di inferenza* da utilizzare per il ragionamento automatico. La sfida del semantic web, quindi, è fornire un linguaggio per esprimere *dati* e *regole* per ragionare sui dati, che consenta l' *esportazione sul web* delle regole da qualunque sistema di rappresentazione della conoscenza.
Esaminando più in dettaglio la Figura 8, si può notare il ruolo di base giocato da **XML** (con *Name Space* e *xmlschema*), che consente di dare ai documenti una *struttura*

arbitraria, mentre **RDF** può essere usato per esprimere il *significato*, asserendo che alcuni particolari elementi hanno delle proprietà (p.es. *autore-di*).
Un terzo componente è l' *Ontology Vocabulary* (*livello ontologico*), inteso come il contenitore che definisce in modo formale le relazioni fra i termini. Una ontologia permette di descrivere le relazioni tra i tipi di elementi (per es. "questa è una proprietà transitiva") senza però fornire informazioni su come utilizzare queste relazioni dal punto di vista computazionale. Le ontologie possono svolgere un ruolo fondamentale nel migliorare il funzionamento del Web (ricerca di concetti, collegamento delle informazioni contenute in una pagina alle strutture di conoscenza associate, etc.).
Il linguaggio definito dal W3C per definire ontologie strutturate, in architettura web, per consentire una migliore integrazione dei dati tra applicazioni in settori diversi è **OWL** (Ontology Web Language).
Il *livello logico* è il livello immediatamente superiore al livello ontologico. A questo livello le asserzioni esistenti sul Web possono essere utilizzate per derivare nuova conoscenza. Tuttavia, i sistemi deduttivi non sono normalmente interoperabili, per cui, secondo Berners-Lee, invece di progettare un unico sistema onnicomprensivo per supportare il ragionamento (*reasoning system*), si potrebbe pensare di definire un linguaggio universale per rappresentare le dimostrazioni. I sistemi potrebbero quindi autenticare con la firma digitale queste dimostrazioni ed esportarle ad altri sistemi che le potrebbero incorporare nel Semantic Web.
La firma digitale (*digital signature*) è di significativa importanza in diversi strati nel modello astratto del Semantic Web. La crittografia a chiave pubblica è una tecnica nota da qualche anno, ma non si è ancora diffusa su larga scala come ci si poteva attendere. Nella visione di Berners-Lee, un elemento che potrebbe aver giocato contro la diffusione di questa tecnica è che essa è a "grana grossa", imponendo una scelta binaria tra fiducia o non fiducia (trusted/not trusted), mentre sarebbe necessaria una infrastruttura in cui le parti possano essere riconosciute e accettate come credibili in specifici domini. Con una granularità più fine come questa, la firma digitale potrebbe essere utilizzata per stabilire la provenienza delle ontologie e delle deduzioni, oltre che dei dati.
L' intera comunità scientifica sta investendo molte energie nel settore del Semantic Web. Molti riferimenti utili si trovano in [SemWeb].

Conclusioni

Nell' evoluzione del web verso il Semantic Web va assumendo sempre maggiore importanza l' interoperabilità, sia tecnologica che semantica. La famiglia di tecnologie XML gioca un ruolo essenziale nei vari livelli architetturali.
A livello di *markup*, XML consente l' *interoperabilità nel contesto delle applicazioni*.
A livello dei *dati*, RDF è l' elemento chiave per l' *interoperabilità tra applicazioni*.
Infine, al livello *ontologico*, linguaggi come OWL consentono di perseguire l' obiettivo del *web of meaning*.

Ringraziamenti

Un doveroso ringraziamento va a tutti quelli che mi hanno aiutato nella preparazione di questo documento, con utili suggerimenti e discussioni chiarificatrici (in particolare, Silvia Martelli e Jeremy J. Carroll). Un particolare ringraziamento va al W3C Team, che

mantiene sul sito documentazione aggiornata. Parte del contenuto di questo lavoro proviene direttamente dal materiale reperibile sul sito.

Bibliografia

[ATAG] *Authoring Tool Accessibility Guidelines 1.0*, http://www.w3.org/TR/WAI-AUTOOLS/

[DC] The Dublin Core Home Page, URL: http://dublincore.org/

[HTML-AF] WAI Resource: HTML 4.0 Accessibility Improvements, http://www.w3.org/WAI/References/HTML4-access

[IRDF] Introduction to RDF Metadata, W3C NOTE 1997-11-13, Ora Lassila, URL:http://www.w3.org/TR/NOTE-rdf-simple-intro

[MathML] W3C's Math Home Page, http://www.w3.org/Math/

[Miller1998] Miller E.: An Introduction to the Resource Description Framework, D-Lib Magazine, May 1998, http://www.dlib.org/dlib/may98/miller/05miller.html

[Naming] Naming and Addressing: URIs, URLs, ..., http://www.w3.org/Addressing/

[RDFMSS] O.Lassila, R.Swick: *Resource Description Framework (RDF) Model and Syntax Specification*,W3C Recommendation 22 February 1999, http://www.w3.org/TR/REC-rdf-syntax

[RDFSS] *Resource Description Framework (RDF) Schema Specification*, W3C Recommendation 03 March 1999, http://www.w3.org/TR/1999/PR-rdf-schema-19990303

[SemWeb] http://www.semanticweb.org/

[Signore2001] Signore O.: *Il ruolo centrale di XML nell' evoluzione del Web*, XML Day Milan, Conference proceedings, Milan, September 21 (find this and other similar papers at: http://www.w3c.it/papers/)

[TBL1997] Tim Berners-Lee: *Metadata architecture*, (1997), http://www.w3.org/DesignIssues/Metadata.html

[TBL1998] Tim Berners-Lee: *Semantic Web Road Map*, (1998), http://www.w3.org/DesignIssues/Semantic.html

[TBL1999] Tim Berners-Lee: *Weaving the Web: The Original Design and Ultimate Destiny of the World Wide Web by Its Inventor*, HarperSanFrancisco (1999), ISBN 0-06-251587-X

[TBL2001] Berners-Lee T., Hendler J., Lassila O.: *The Semantic Web*, Scientific American, May 2001, http://www.scientificamerican.com/2001/0501issue/0501berners-lee.html

[TUUAG] *Techniques for User Agent Accessibility Guidelines*, http://www.w3.org/WAI/UA/WAI-USERAGENT-TECHS/

[UAAG] *User Agent Accessibility Guidelines 1.0*, http://www.w3.org/TR/WAI-USERAGENT/

[W3C] *World Wide Web Consortium Home Page*, http://www.w3.org

[WA-Policies] *Policies Relating to Web Accessibility*, http://www.w3.org/WAI/Policy/

[WCAG] *Web Content Accessibility Guidelines 1.0*, http://www.w3.org/TR/WAI-WEBCONTENT/

[XML] *Extensible Markup Language (XML)* , http://www.w3.org/XML/

[XML1.0] *Extensible Markup Language (XML) 1.0 (Second Edition) W3C Recommendation 6 October 2000*, http://www.w3.org/TR/2000/REC-xml-20001006

[XMLns] *Namespaces in XML* - World Wide Web Consortium 14-January-1999
http://www.w3.org/TR/REC-xml-names/

[XMLschema0] XML Schema Part 0: Primer - W3C Recommendation - 2 May 2001
http://www.w3.org/TR/xmlschema-0/

[XMLschema1] XML Schema Part 1: Structures - W3C Recommendation - 2 May 2001
http://www.w3.org/TR/xmlschema-1/

[XMLschema2] XML Schema Part 2: Datatypes - W3C Recommendation - 2 May 2001
http://www.w3.org/TR/xmlschema-2/