

## Integrare informazioni e applicazioni nel KMS con i web service

*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](mailto: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*.

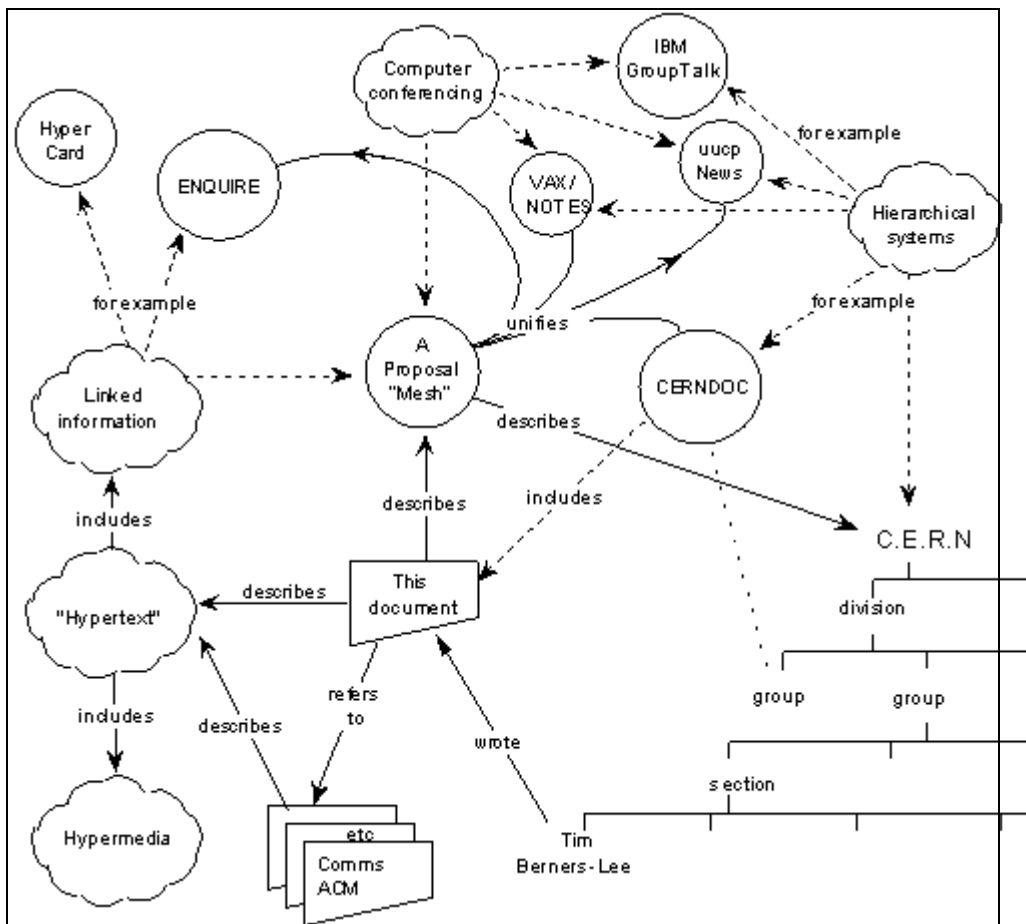


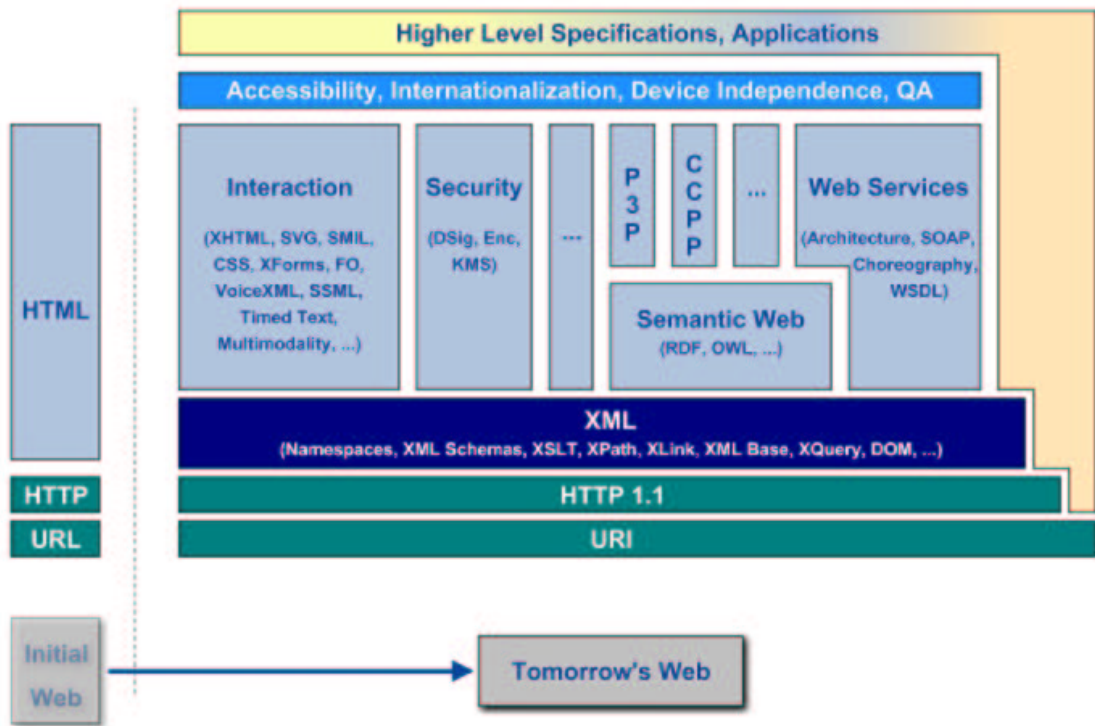
Figura 1 - La visione originaria del World Wide Web

## Il Web

### 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*).



**Figura 2 - L'evoluzione del 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 ha ereditato i principi fondamentali di Internet, cioè *interoperabilità, evoluzione e decentralizzazione*.

In questo lavoro ci occuperemo essenzialmente degli aspetti di interoperabilità, che, nel contesto attuale, costituisce un fattore chiave di successo. Due applicazioni vengono dette interoperabili se si possono scambiare dati e servizi in modo efficace e consistente, permettendo la comunicazione tra piattaforme hardware e software eterogenee. 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 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.

## L'interoperabilità tecnologica

### Strutturazione dell'informazione

**XML** – Alla base dell'interoperabilità tecnologica c'è la possibilità di strutturare l'informazione, in modo che sia comprensibile e trasferibile in modo agevole. *Extensible Markup Language (XML)* è nato per far fronte alle limitazioni di HTML nella realizzazione delle nuove applicazioni Web, in cui i dati costituiscono 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”.

**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<sup>1</sup> 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 si basano sulla comprensione del principio chiave di *separazione tra contenuto e forma*. L' architettura di riferimento prevede che le informazioni, estratte dalla base dati aziendale, vengano strutturate in un documento XML, che viene successivamente trasformato nel formato più adatto per l' utente finale, mediante una trasformazione di stile. Si ricordi 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.

**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.).

SMIL, MathML e SVG sono alcune tecnologie, basate su XML, che consentono di arricchire i contenuti testuali.

**Synchronized Multimedia Integration Language (SMIL**, pronunciato come la parola inglese "smile") permette in modo semplice la creazione (“authoring”) di presentazioni interattive audiovisive.

HTML presenta diverse limitazioni nella rappresentazione e presentazione di formule matematiche. MathML ([MathML]) è un linguaggio di marcatura che permette di scrivere formule matematiche anche molto complesse. 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.

---

<sup>1</sup> Un **XML namespace** è un insieme di nomi, caratterizzati da un URI di riferimento, utilizzati come element type e attribute name.

Per la grafica vettoriale esiste la specifica SVG (**Scalable Vector Graphics**). SVG è un linguaggio per descrivere grafici bidimensionali in XML. 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.

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 ha una valenza non solo *sociale*, ma anche *economica* (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 il principio dell' 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**

### **Un problema tipico**

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, e 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.

## 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 dal fatto che i metadati sono comprensibili dalla macchina (*machine understandable*), e quindi costituiscono un tipo di informazione che può essere utilizzata dai *software agent*, per fare un uso appropriato delle risorse. 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 perciò possono essere *memorizzati come dati*, (nella risorsa stessa o in un’altra risorsa), e possono essere *descritti da altri metadati*, e così via. Il numero di metalivelli da specificare dipende dalle caratteristiche delle applicazioni e dalle specifiche esigenze.

## II Resource Description Framework

I *metadati* sembrano offrire una soluzione al problema di automatizzare il Web restando ancorati alla sua architettura originaria, in cui tutte le informazioni erano *machine-readable*, ma non *machine-understandable*. L’uso efficace dei metadati, tuttavia, richiede che vengano stabilite delle convenzioni per la *semantica*, la *sintassi* e la *struttura*. Coerentemente con l’assunzione di fondo di una architettura peer-to-peer, le *single 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 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).



- Properties* Una *property* (proprietà) è un aspetto specifico, una caratteristica, un attributo, o una relazione utilizzata per descrivere una risorsa. 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 altro tipo primitivo definito da XML) oppure un'altra risorsa.

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. Uno degli esempi più noti è la Dublin Core Initiative ([DC]). 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.

## Lo scenario del Semantic Web

La sfida attuale, su cui l'intera comunità scientifica sta investendo molte energie ([SemWeb]), è il Semantic Web, la cui architettura a livelli è riportata in Figura 3.

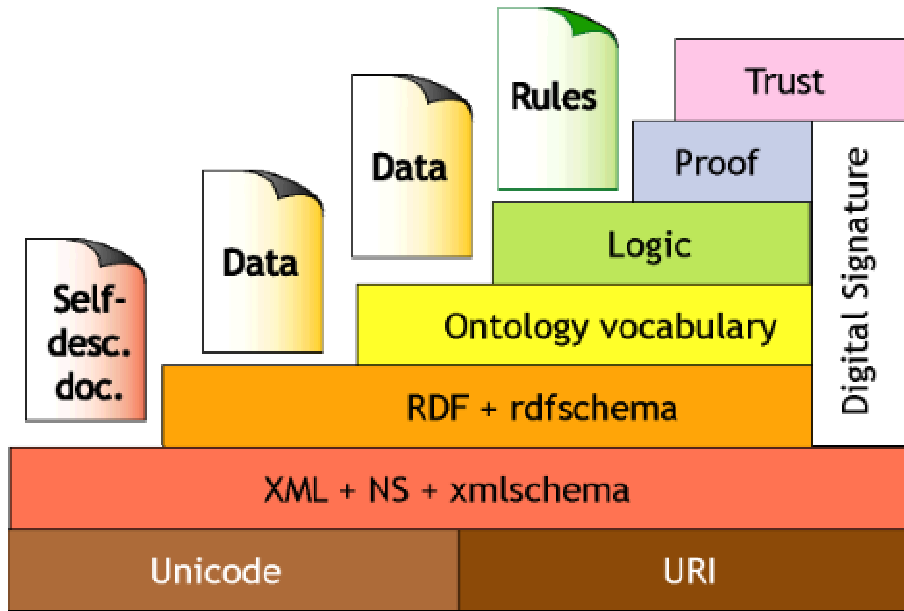
Va ricordato che la filosofia di base del Web è quella di uno spazio informativo universale, navigabile, con un mapping da *URI (Uniform Resource Identifier)* alle risorse. Nel contesto del Semantic Web, il termine semantico assume la valenza di "*elaborabile dalla macchina*". Il Semantic Web è, come 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.

Il Semantic Web ([TBL2001]) 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.

Lo schema in Figura 3 evidenzia 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, 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.).



**Figura 3 - L' architettura del Semantic Web**  
<http://www.w3.org/2000/Talks/1206-xml2k-tbl/slide10-0.html>

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.

La firma digitale (*digital signature*) è di significativa importanza in diversi strati nel modello astratto del Semantic Web. È necessaria infatti una infrastruttura in cui le parti possano essere riconosciute e accettate come credibili in specifici domini. Con una granularità fine come questa, la firma digitale potrebbe essere utilizzata per stabilire la provenienza delle ontologie e delle deduzioni, oltre che dei dati.

## Uno scenario di interoperabilità: i Web Service

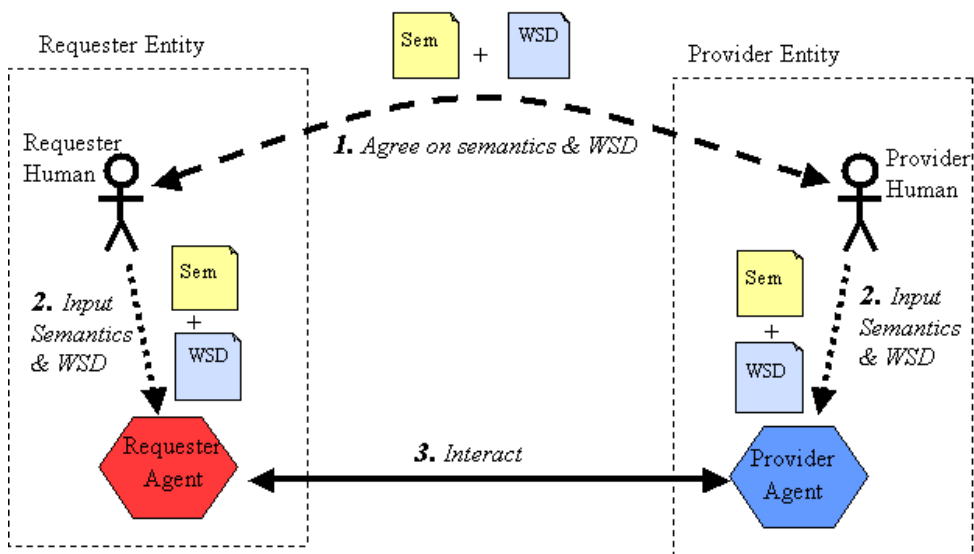
### Definizione e architettura di un Web Service

Una possibile definizione di WEB Service è la seguente:

*A Web service is a software system designed to support interoperable machine-to-machine interaction over a network. It has an interface described in a machine-processable format (specifically WSDL).*

Other systems interact with the Web service in a manner prescribed by its description using SOAP-messages, typically conveyed using HTTP with an XML serialization in conjunction with other Web-related standards.

**Agenti e Servizi.** - Un Web Service è una nozione astratta che deve essere implementata da un agent reale. L' *agent* è quindi l' *entità concreta* (il software) che spedisce e riceve messaggi, mentre il *service* è l' *insieme astratto di funzionalità fornite*. L' obiettivo di un Web Service è fornire alcune funzionalità per conto del suo proprietario (una persona o un' organizzazione). La *provider entity* è la persona o organizzazione che fornisce un idoneo agent per implementare un particolare servizio (vedi Figura 4).



**Figura 4 - I vari ruoli nell' architettura dei Web Service (da [WSA])**

Una *requester entity* è una persona o organizzazione che intende utilizzare il web Service di una *provider entity*. A tale scopo, utilizzerà un *requester agent* per scambiare messaggi con il *provider agent* della *provider entity*. Questo scambio di messaggi potrà avere successo solo se *provider* e *requester entity* si saranno accordate sulla *semantica* e la *meccanica* dello scambio di messaggi, documentati in una *Web Service Description (WSD)*. La *WSD* è una specifica *machine processable* dell' interfaccia, e definisce formato, datatype, protocollo di trasporto, locazioni di rete ove può essere invocato il servizio, etc.

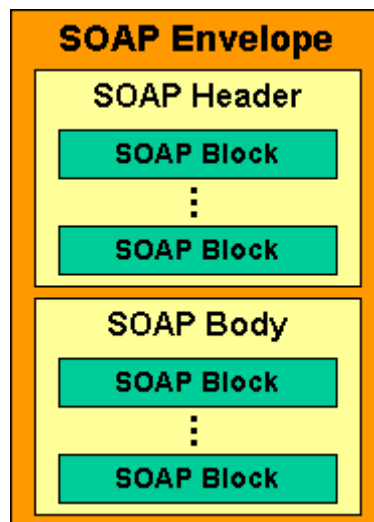
La *semantica* dello scambio di messaggi rappresenta il "contratto" tra *requester* e *provider entity* riguardo allo scopo e conseguenze dell' interazione

In altri termini, la descrizione del servizio (*WSD*) rappresenta un contratto che governa l' *aspetto meccanico* dell' interazione con un particolare servizio, la *semantica* rappresenta un contratto che regola il *significato e lo scopo* dell' interazione.

**Il ruolo degli operatori umani.** - Sebbene uno degli obiettivi principali dei Web Service sia l' automazione di processi che altrimenti sarebbero manuali, la componente umana gioca ancora un ruolo fondamentale, principalmente in due modi:

1. Gli esseri umani devono concordare sulla semantica e sulla descrizione dei servizi. Considerato che alla fin fine è sempre un essere umano (o una organizzazione) a detenere il possesso di un Web Service, le persone devono raggiungere, implicitamente o esplicitamente, un *accordo sulla semantica e sulla descrizione del servizio che governa l'interazione*.
2. Gli esseri umani direttamente o indirettamente creano i requester e provider agent, e, in definitiva, devono garantire che questi agenti *implementino la descrizione e la semantica del servizio concordato*.

Indipendentemente dall'approccio adottato, dal punto di vista informativo è necessario che la semantica e la descrizione del servizio vengano in qualche modo forniti o incorporati nei due agenti prima che essi possano interagire.



**Figura 5 – Struttura di un messaggio SOAP**

**SOAP.** - In estrema sintesi, SOAP 1.2 ([SOAP12]) fornisce la definizione di informazione XML-based che può essere utilizzata per scambiare informazione strutturata e tipata tra componenti paritarie in un ambiente decentralizzato e distribuito. SOAP è fondamentalmente un paradigma di scambio messaggi stateless e unidirezionale, ma le applicazioni possono creare modelli di interazione più complessi.

Un messaggio SOAP è composto come in Figura 5: una *envelope* che contiene due elementi: un Header (opzionale) e un Body.

L' *Header* è opzionale, ed è stato progettato in previsione dei vari utilizzi di SOAP, molti dei quali potrebbero coinvolgere altri nodi (gli intermediari SOAP) che elaborino il messaggio SOAP nel suo cammino dal mittente al destinatario finale. I figli di primo livello del SOAP Header sono gli *header block*, che rappresentano raggruppamenti logici di dati che possono essere indirizzati a intermediari previsti nel cammino del messaggio. Il *Body*, con i suoi figli, è obbligatorio, e contiene l'informazione principale da trasmettere tra il mittente e il destinatario, e quindi si suppone implicitamente che il contenuto dei suoi figli (i *body block*) sia comprensibile ed elaborabile dal destinatario finale.

## Un esempio

Lo scenario d'uso di Figura 6 descrive come un utente potrebbe prenotare un pacchetto vacanze (volo e pernottamento).

Questo scenario d'uso presuppone che tutti i servizi facciano riferimento a concetti comuni (volo, classe economica, pernottamento, etc.). Perché il servizio dell'agenzia di viaggi possa comprendere i servizi della compagnia aerea e inviare l'informazione corretta, è necessario che sia stata definita, e venga usata sul Web, una *ontologia per il turismo* (Figura 7). Ricordiamo che un'ontologia è una descrizione formale di un insieme di concetti e delle loro relazioni, quindi, associando un nome ad ogni concetto, un'ontologia definisce il vocabolario standard da utilizzare per comunicare questi concetti.

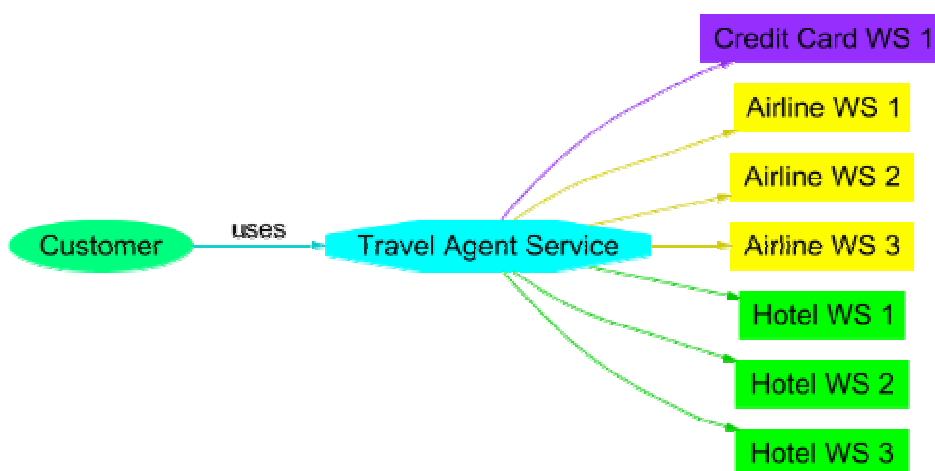


Figura 6 - Uno scenariod'uso (da [WSA-US])

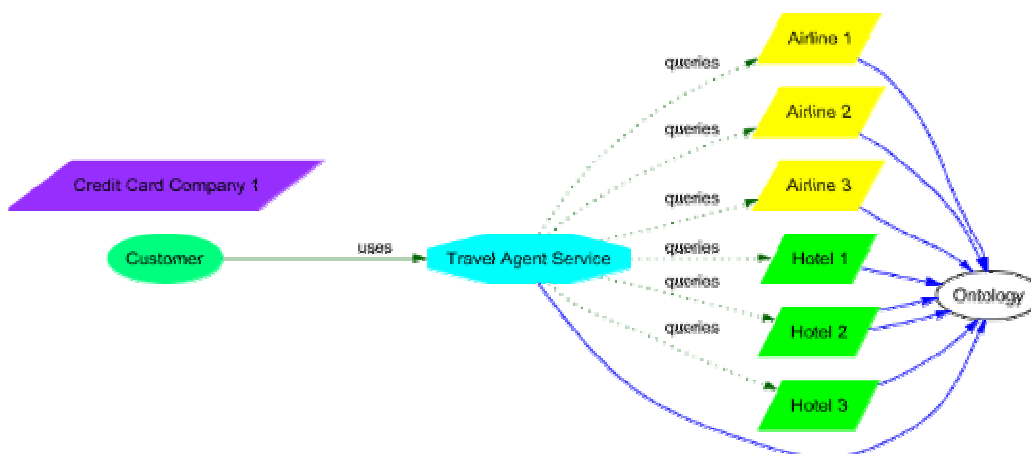


Figura 7 – Il ruolo di un'ontologia in uno Scenario d'uso (da [WSA-US])

## 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*.

## 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/>
- [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>
- [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>
- [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>
- [WCAG] *Web Content Accessibility Guidelines 1.0*, <http://www.w3.org/TR/WAI-WEBCONTENT/>
- [WSA] *Web Services Architecture* - W3C Working Draft 8 August 2003,

	<a href="http://www.w3.org/TR/ws-arch/">http://www.w3.org/TR/ws-arch/</a>
[WSA-US]	Web Services Architecture Usage Scenarios - W3C Working Draft 14 May 2003, <a href="http://www.w3.org/TR/ws-arch-scenarios/">http://www.w3.org/TR/ws-arch-scenarios/</a>
[XML]	<i>Extensible Markup Language (XML)</i> , <a href="http://www.w3.org/XML/">http://www.w3.org/XML/</a>
[XML1.0]	<i>Extensible Markup Language (XML) 1.0 (Second Edition) W3C Recommendation 6 October 2000</i> , <a href="http://www.w3.org/TR/2000/REC-xml-20001006">http://www.w3.org/TR/2000/REC-xml-20001006</a>
[XMLns]	<i>Namespaces in XML</i> - World Wide Web Consortium 14-January-1999 <a href="http://www.w3.org/TR/REC-xml-names/">http://www.w3.org/TR/REC-xml-names/</a>
[XMLschema0]	XML Schema Part 0: Primer - W3C Recommendation - 2 May 2001 <a href="http://www.w3.org/TR/xmlschema-0/">http://www.w3.org/TR/xmlschema-0/</a>
[XMLschema1]	XML Schema Part 1: Structures - W3C Recommendation - 2 May 2001 <a href="http://www.w3.org/TR/xmlschema-1/">http://www.w3.org/TR/xmlschema-1/</a>
[XMLschema2]	XML Schema Part 2: Datatypes - W3C Recommendation - 2 May 2001 <a href="http://www.w3.org/TR/xmlschema-2/">http://www.w3.org/TR/xmlschema-2/</a>
[XSL]	<i>Extensible Stylesheet Language (XSL)</i> , Version 1.0 W3C Recommendation 15 October 2001 <a href="http://www.w3.org/TR/2001/REC-xsl-20011015/xslspecRX.pdf">http://www.w3.org/TR/2001/REC-xsl-20011015/xslspecRX.pdf</a>
[XSLT]	<i>XSL Transformations (XSLT)</i> Version 1.0, W3C Recommendation 16 November 1999, <a href="http://www.w3.org/TR/xslt">http://www.w3.org/TR/xslt</a>