# FORMATO DEI DATI

Per documentare/aggiungere metadati -> per esempio paolo rossi ha un ruolo. Per farlo possiamo usare i tag. Parliamo dei tag language:

dominio=contesto dell’informazione, scope del metodo

distinguo i tag in procedurale (mi dice anche cosa ne devo fare) oppure descrittivo (dice solo che cos’è il metadato)

che cosa è hypermedia control, e cosa fa? applica cambiamenti di stato usando i link

hypermedia invece sono collegamenti ipertestuali

### XML

XML = scambio di informazioni semistrutturate. XML è più complesso di HTML per questo non ha preso piede, ma ora HTML5 ha ereditato molte cose del XML. I tag non è fisso, ma è libero. Rossi ha diversi ruoli. C’è flessibilità però va in contrasto con l’affidabilità.

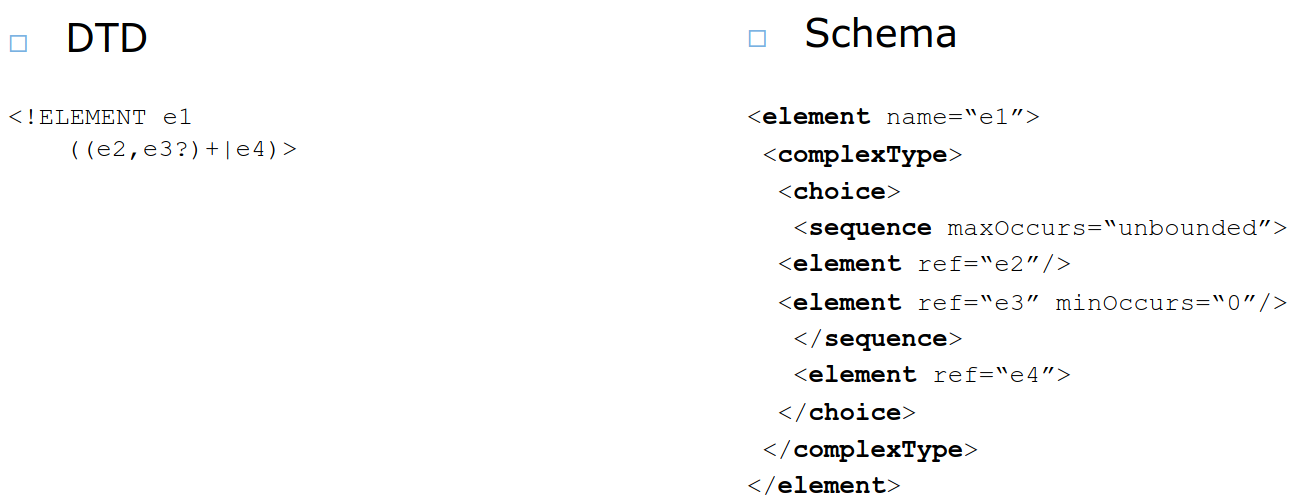
Infatti in HTML5 i tag sono fissi, quindi è un pò diverso da XML. DTD definisce la grammatica di XML.

<ruolo> Paolo Rossi </ruolo>

un documento XML o XML object è well-formed (la sintassi è corretta) e valido (rispettare una certa grammatica).

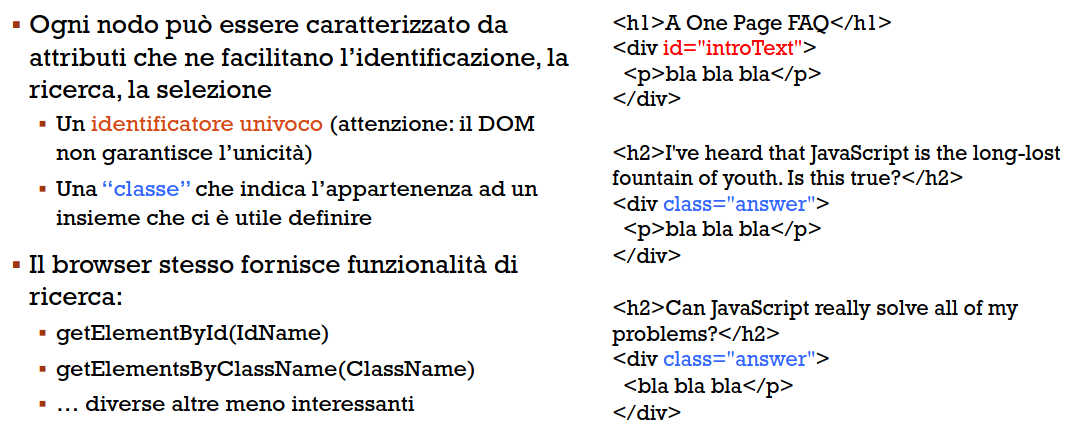
DTD dice se un documento è valido. Dice se devo avere un titolo, autore, etc… Se creo un file File.dtd e in quel file ci scrivo le regole della grammatica (come in LC). Ci sono dei limiti: DTD non può dire se la lunghezza di un elemento non può essere negativo, i commenti possono apparire ovunque.

Per fare tutto ciò hanno trovato una soluzione. XML Schema. È XML dedicato per la validazione di una pagina/documento. Posso definire namespaces, i nomi dei tag possono essere usati in contesti privati. XML Schema è estremamente complicato.

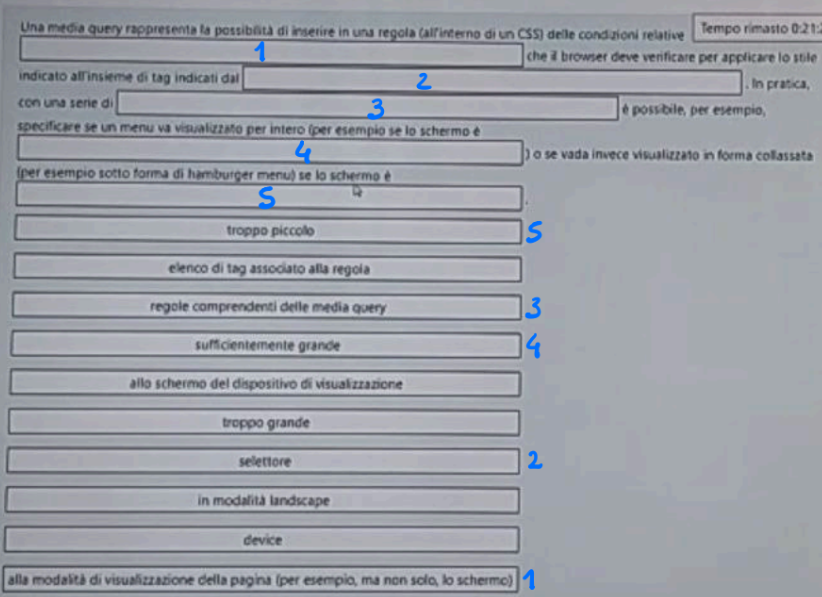


### HTML

HTML, è un linguaggio di markup per la specifica di contenuti e struttura di pagine web.

DOM (Document Object Model) = interfaccia neutrale/API che gestisce dinamicamente la struttura(HTML) e lo stile(CSS) di un documento web. 

Le media Query invece sono selettori (simili a class, id) che sono capaci di capire che tipo di medium sta interagendo con il sito web e definire delle regole stilistiche. *Per esempio capire se un è telefono o un PC e modificare la struttura della pagina di conseguenza.*



Per gestire i conflitti tra diversi stylesheet e decidere quale stile applicare a un elemento:

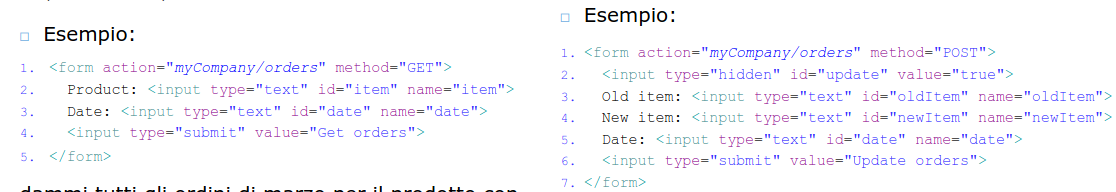
* **Importanza**: Questo flag specifico può essere aggiunto a un attributo CSS per aumentarne l'importanza. Queste regole sovrascrivono qualsiasi altra regola.
* **Specificità**: La specificità è un algoritmo utilizzato dai browser per determinare quale dichiarazione CSS è la più rilevante per un elemento. Viene calcolata in base al numero di selettori di ciascuna categoria (ID, CLASS, TYPE) nel selettore che corrisponde all'elemento. Se 2 dichiarazioni sono uguali, si usa il terzo metodo.
* **Ordine nel sorgente**: Se 2 dichiarazioni hanno la stessa importanza, l'ultima dichiarazione nel sorgente CSS avrà la precedenza. Questo significa che la dichiarazione che appare per ultima nel codice CSS avrà effetto sull'elemento.

La definizione di uno stile CSS implica la conoscenza della struttura del documento HTML.

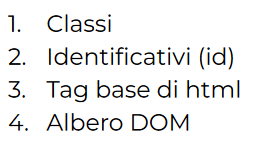
Chi fa la struttura NON deve sapere NECESSARIAMENTE anche lo stile

Elemento base di un CSS = Regola che specifica un selettore e una serie di coppie attributo-valore.

### FORM

Nei form c’è una action (che è un tag). Vediamo un esempio con GET (dati in coda al URL) e con POST (dati nel body, più sicuro).

Javascript definisce e governa la dinamicità della pagina, il modo della pagina di cambiare in modo reattivo al comportamento dell’utente o anche proattivo, senza necessariamente dover attendere uno stimolo dall’utente



### JSON

È un formato per lo scambio dei dati. Sono coppie di dati semplificati. Più leggero rispetto a XML. Infatti non è un mark-up language, non permette di formattare un documento, non ha le funzioni.

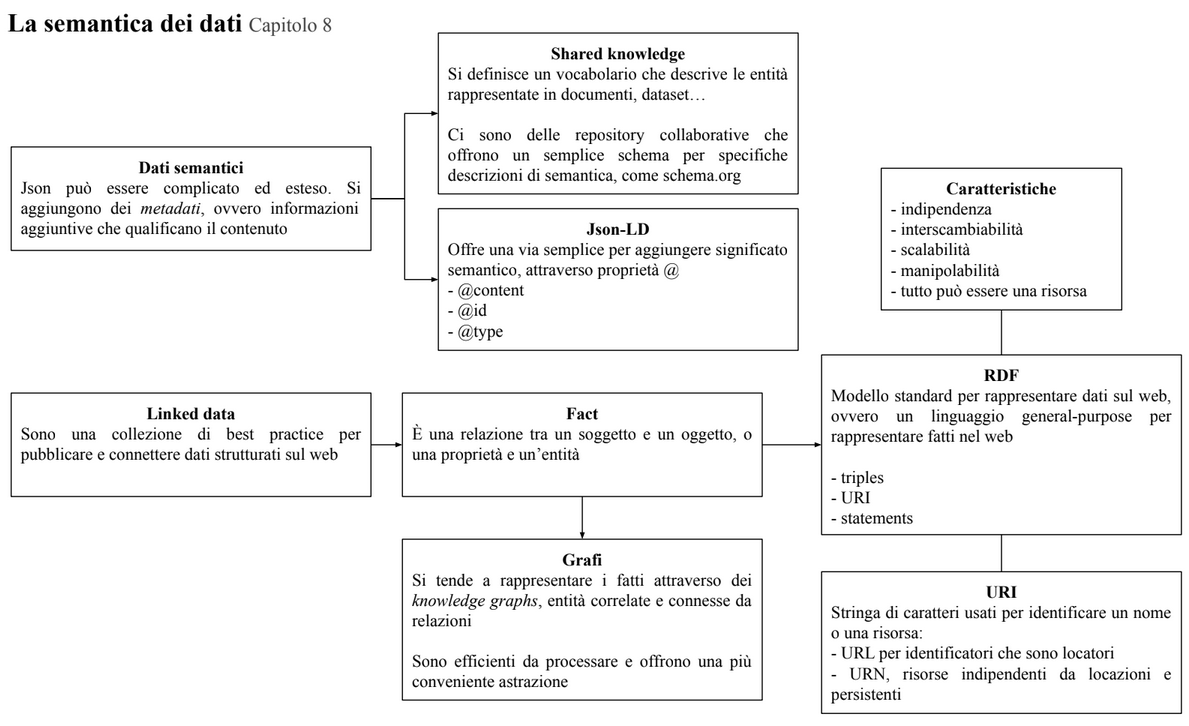
JSON non ha versioni, quindi è molto stabile.

Le graffe indicano gli oggetti, le parentesi quadre indicano gli array. Gli attributi sono dati da “attributo”. I tipi base sono Number, String, Boolean, Null. Può accettare testo non-JSON.

Questo non permette a JSON di rappresentare oggetti Java.

Javascript è un superset di JSON, ovvero una versione estesa.

# DATI SEMANTICI



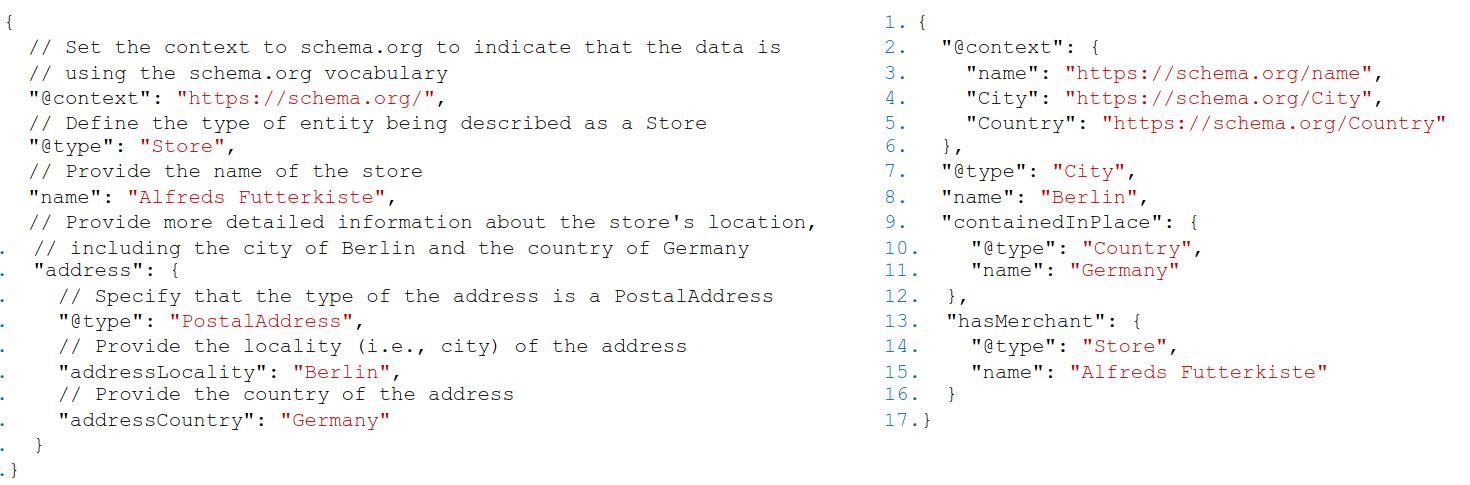
Il problema della semantica è aggiungere **contesto ai dati**. HTML non è abbastanza per definire un vocabolario condiviso, in grado di descrivere elementi in una pagina in modo non ambiguo.

JSON permette di condividere i dati, ma non c’è contesto. **JSON-LD** è la soluzione.

È un’estensione del JSON dove si aggiungono URL che definiscono uno ***schema***particolare, identificatori creati da URI e URL che definiscono i tipi di dati. Tutte queste aggiunte condividono il fatto di iniziare con la @.

Una **conoscenza** può avere diversi livelli di dettaglio: **vocabolario**, **grafo** e **ontologia**. Esse vengono condivise da più sistemi.

Come vengono mantenute, sono spesso repository condivisi che tutti possono accedere. Wikidata è la base di conoscenza più usata. Un’altro è Schema.org.



Definisco prima il contesto, i tipi e altro. Solo dopo scrivo il JSON.

* **@context** dice che sti sta riferendo al vocabolario di uno schema in particolare
* **@id** è un identificatore univoco di un URI
* **@type** definisce il tipo
* **@vocab** : :<https://schema.org/>” -> dice che il vocabolario base è quello
* **wd:Q64** -> vuol dire che l’id fa anche riferimento a wikidata

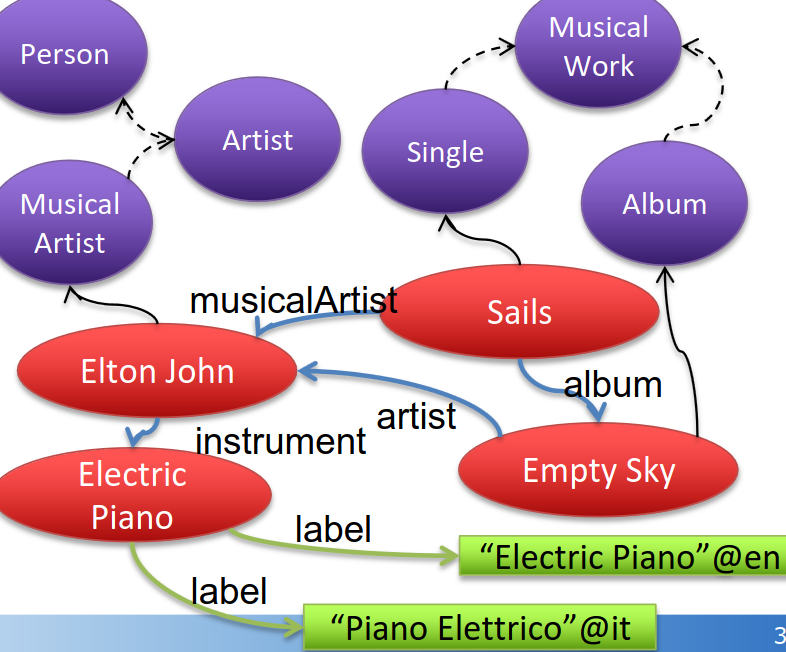


JSON rappresenta una Persona. JSON-LD identifica una persona. Le API condividono messaggi di *@tipe* persona.

### COME CONDIVIDERE LA CONOSCENZA

Esisteva un Linked Open Data Cloud. Erano i vari Wikidata, DBpedia, etc…

Usiamo i grafi, perché sono molto efficienti, vediamo un esempio:



I grafi possono essere molto "scalabili" dando la possibilità di aggiungere tantissime informazioni. A loro volta esistono intersezioni di grafi.

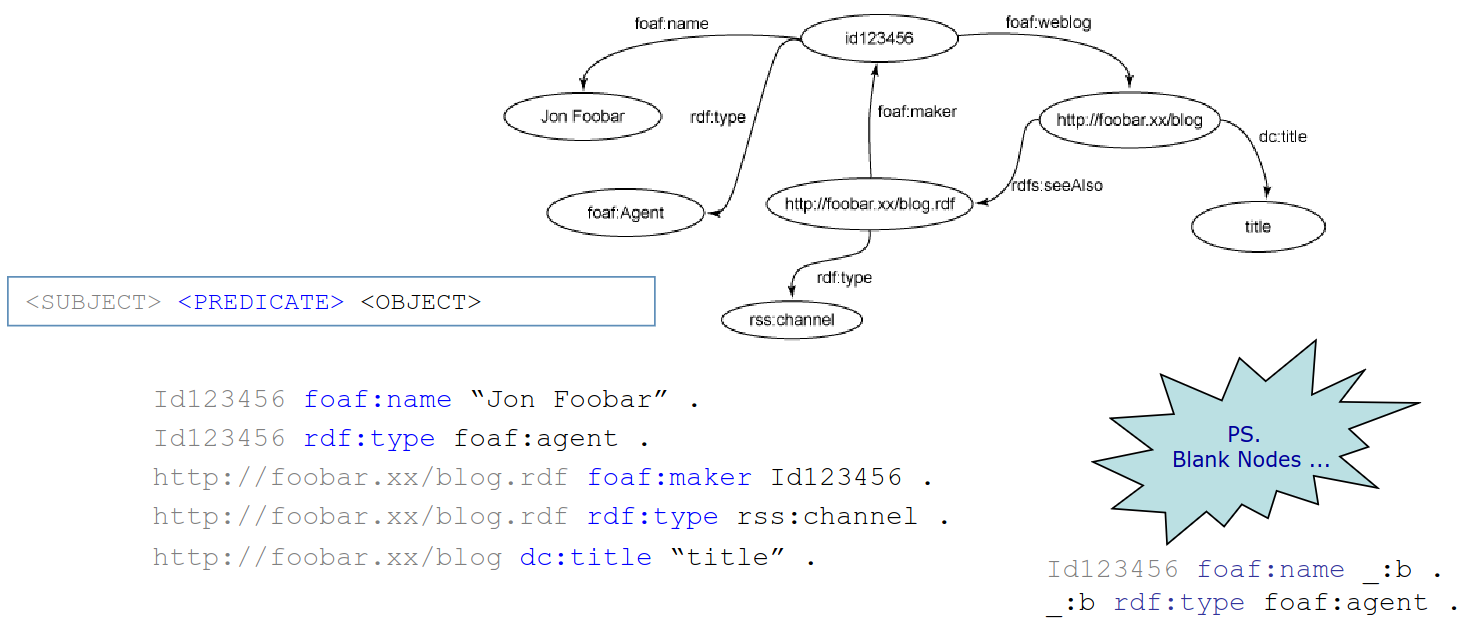
A volte i dati sono pure dinamici, tipo quando cerco su google la popolazione di una città, esso è un dato che è in costante cambiamento.

### RDF - Resource Description Framework

È un linguaggio per rappresentare fatti sul web.

L’unità informativa alla base di un RDF è una tripla composta da soggetto, predicato, oggetto (del quale può essere un tipo di dato predefinito).

In RDF ogni cosa è una risorsa e tutto deve essere identificabile tramite URI a meno di non essere un dato primitivo oppure un nodo “blank”.



insieme di triple RDF è caratterizzabile come orientato ed etichettato.

FOAF e DC sono esempi di vocabolario (insieme di concetti e rappresentazioni di conoscenze specifiche) ai quali è associabile un prefisso/abbreviazione che evita di dover scrivere un uri per esteso. Si usa semplicemente perchè è più comodo.

Per esempio -> foaf:name "John Foobar”

