

# RIA

Tuesday, 9 May 2023

10:54

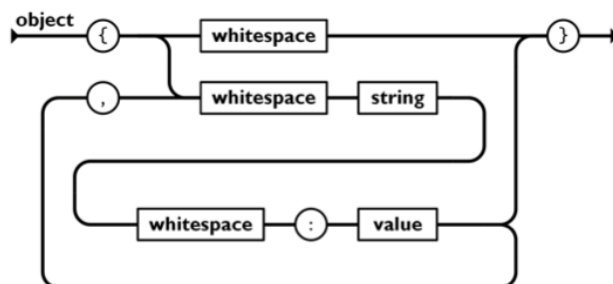
- Rich internet application (Interazione con il server)

- Esempio:

- `Var xhttp = new XMLHttpRequest();` -> Creo messaggio http, aka socket  
`// Faccio l'overwrite della variabile "onreadystatechange", che verrà chiamata d`  
`Xhttp.onreadystatechange = function() {`  
`/*`  
`readyState == 0 -> open() non è ancora stata creata`  
`== 1 -> open() è stata chiamata`  
`== 2 -> send() è stata chiamata`  
`== 3 -> stà scaricando, responseText è parziale`  
`readyState == 4 -> l'ho scaricato completamente, operazione completata`  
`*/`  
`// status == 200 -> l'ho ricevuto completamente`  
`If (xhttp.readyState == 4 && xhttp.status == 200 {`  
`// responseText = body della risposta`  
`Console.log(xhttp.responseText); //responseXML esiste`  
`}`  
`}`  
`Xhttp.open("GET", "home/ajax_text.txt", true);` -> Creo un messaggio, richiedo l  
`Xhttp.send();`

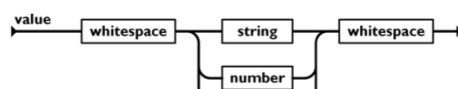
- JSON

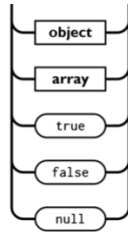
- Usato per lo scambio di dati
- Ha una struttura generale che segue la seguente immagine:



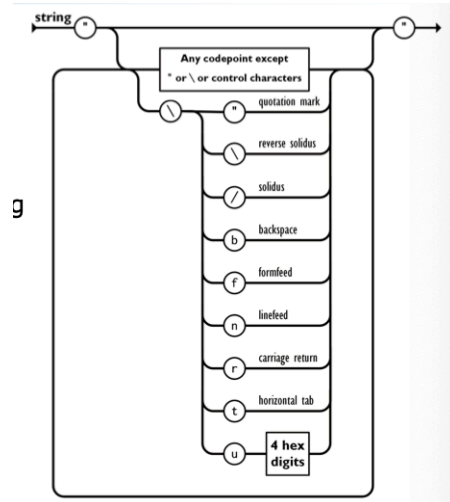
E nello specifico, dentro value abbiamo:

- Value

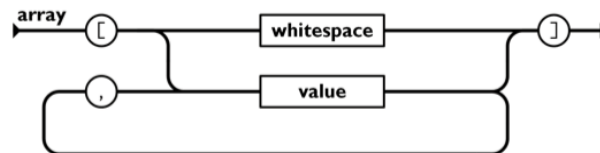




## ■ String



## ■ Array



- JSON.parse(text) -> parsing
- Per inviare json bisogna fare `setRequestHeader("content-type", "application/x-v")`
- Un problema del json è che, non sappiamo il contesto dei dati.

Che tipo è? A cosa si riferisce?

Ed è questo che è nato JSON-D

Aggiungere i tipi, contesto ed altri dati alle informazioni.

### ■ Keyword è @

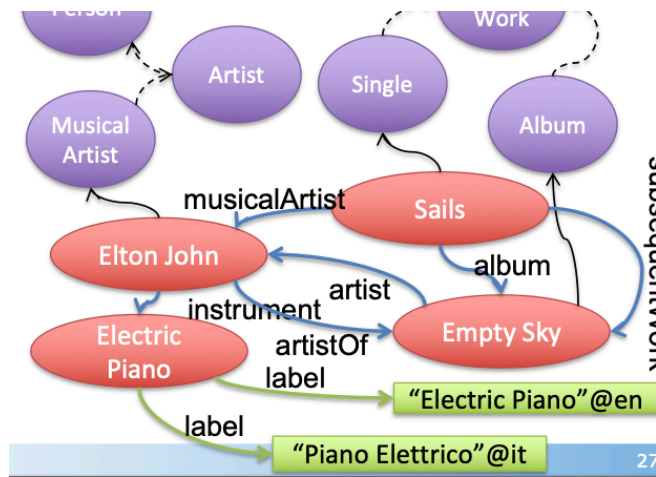
Ed abbiamo 3 key:

- ☐ Context -> Riferenza ad uno schema particolare
- ☐ Id -> unique identifier
- ☐ Type -> tipo del valore, una classe. Tipo "Città"  
L'ho interpretato come un context custom

## - Linked data

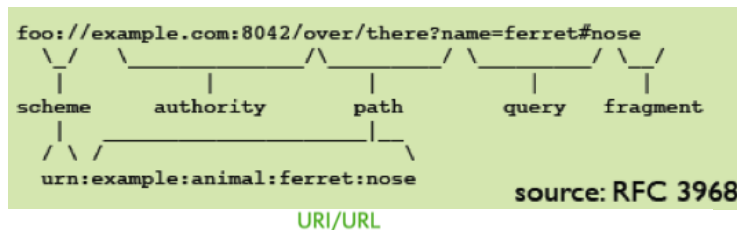
- E' una serie di "best practice" per connettere strutture date del web
- Indichiamo dov'è la prossima risorsa, aka facciamo dei collegamenti
  - Il web è tipo una linked data attraverso i link
- Ogni entità è connessa attraverso le altre con "tipo" e "classe di"





## - RDF

- E' un modello per rappresentare dati nel web, ed è basato su:
  - Triplette, praticamente unità basiche per organizzare le informazioni
  - Drected graphs, insieme di triplette
  - URI, identificatore
    - E' una stringa usata per identificare il nome di una risorsa
    - Formato da:
      - ◆ URL  
Aka a chi ci riferiamo
      - ◆ URN  
Aka la risorsa che vogliamo
- Un modo per rappresentare dati è con la sintassi XML



- Caratteristiche:
  - Indipendente
  - Interscambiabile
  - Scalabile
  - Le proprietà sono risorse

- Le proprietà sono risorse
- I soggetti e gli oggetti sono risorse