

# XML

1

**ALBERTO BELUSSI**  
**ANNO ACCADEMICO 2011/2012**

# XML

2

XML è un linguaggio di marcatura proposto dal W3C

- XML definisce una sintassi generica per contrassegnare i dati di un documento elettronico con marcatori (tag) semplici e leggibili

La sintassi XML viene utilizzata in contesti molto diversi:

- pagine web
- **scambio di dati tra applicazioni web**
- grafica vettoriale
- cataloghi di prodotti
- sistemi di gestione di messaggi vocali
- ...

Riferimento: <http://www.w3c.org/XML>

- “Extensible Markup Language (XML) is a simple, very flexible text format derived from SGML (ISO 8879). Originally designed to meet the challenges of large-scale electronic publishing, XML is also playing an increasingly important role in **the exchange of a wide variety of data** on the Web and elsewhere.”

# XML evoluzione

3

## 1986: Standard Generalized Markup Language (SGML)

- Linguaggio di marcatura strutturato, per la rappresentazione elettronica di documenti di tipo testuale

## 1995: HyperText Markup Language (HTML)

- Applicazione di SGML che permette di descrivere come il contenuto di un documento verrà presentato

## 1998: eXtensible Markup Language (XML)

- Versione “leggera” di SGML, che consente una formattazione semplice e molto flessibile.

# HTML vs XML

4

## HTML

- Insieme **fisso** di tag
- Descrizione degli aspetti di **presentazione** del documento
- Usato solo per la costruzione di **pagine web**

## XML

- Insieme **non fisso** di tag: i tag possono essere personalizzati
- Descrizione del **contenuto informativo** del documento
- Usato in molti **domini diversi**

# HTML vs XML

5

```
<h1>
  Essential XML: <br>
    Oltre il Markup
</h1>

<ul>
  <li>
    Don Box
  </li>
  <li>
    Aaron Skonnard
  </li>
  <li>
    John Lam
  </li>
</ul>

<i>
  Addison-Wesley
</i>
```

```
<titolo>
  Essential XML:
    Oltre il Markup
</titolo>

<autore>
  Don Box
</autore>
<autore>
  Aaron Skonnard
</autore>
<autore>
  John Lam
</autore>
<casa editrice>
  Addison-Wesley
</casa editrice>
```

# Sintassi base di un documento XML

6

## Documenti “**ben formati**”

XML è più restrittivo di HTML per quanto riguarda il posizionamento dei tag e il modo in cui vengono scritti

Ogni documento XML deve essere **ben formato**, vale a dire:

- Deve avere una sola radice
- L'annidamento dei tag deve essere corretto
- Tutti i tag aperti devono essere chiusi
- I valori degli attributi devono essere specificati tra virgolette

# Esempio di documento XML

7

Un documento XML è un file di testo

Esempio persona.xml contiene:

```
<!-- Dati Persona -->
```

**commento**

```
<persona cod_fiscale="RSSMRA65E25L781T">
```

**tag  
iniziale**

```
<nome>
```

```
Luca Rossi
```

**attributo**

**elemento**

**tag  
finale**

```
</nome>
```

**Contenuto  
strutturato**

```
<indirizzo>
```

```
<via> Via Mazzini </via>
```

```
<civico> 10 </civico>
```

```
<città> Verona </città>
```

```
</indirizzo>
```

**elemento**

```
</persona>
```

# Sintassi base di un documento XML

8

Il documento `persona.xml` contiene un elemento radice: `<persona>`

L'elemento `<persona>` ha una struttura: contiene un elemento `<nome>` e un elemento `<indirizzo>`.

L'elemento `<nome>` contiene un testo.

L'elemento `<indirizzo>` ha invece una struttura ulteriore: contiene gli elementi: `<via>`, `<civico>` e `<città>`.



# Sintassi di un elemento XML

9

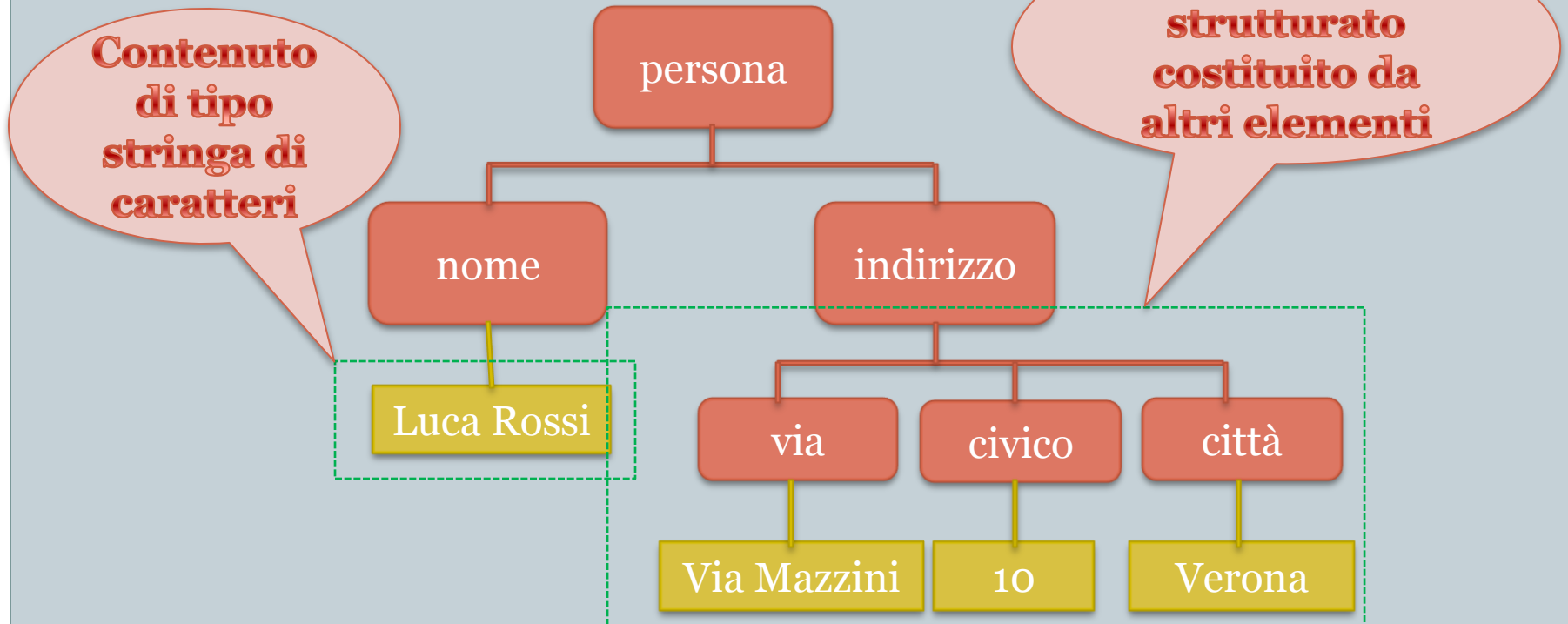
Ogni elemento è caratterizzato da:

- Una marca iniziale (o tag iniziale)
  - `<nome_elemento>`
- Una marca finale (o tag finale)
  - `</nome_elemento>`
- Un contenuto: può essere un valore atomico o un valore strutturato attraverso altri elementi XML (elementi figli).  
Gli elementi vuoti: sono elementi privi di contenuto
  - Esempio: `<età></età>`
- Un elemento può essere opzionale (struttura variabile)
- XML è sensibile alla differenza tra maiuscole e minuscole (è *case-sensitive*)

# Sintassi di un elemento XML

10

Ogni documento XML è un albero di elementi. Ad esempio l'albero del documento **persona.xml** può essere rappresentato come segue:



# Sintassi di un elemento XML

11

Il contenuto di un elemento può anche essere **misto**, come nel seguente esempio:

```
<dati_anagrafici>Il signor  
  <persona>  
    <nome>Mario Rossi</nome> vive in  
    <indirizzo>  
      <via>Via Mazzini</via>  
      <civico>10</civico>  
      <città>Verona</città>  
    </indirizzo>  
  </persona>  
</dati_anagrafici>
```

# Sintassi di un elemento XML

12

Un elemento può essere corredato di attributi.

- Un attributo consiste in una coppia nome-valore associata al tag iniziale di un elemento
  - Ad esempio: `<persona cod_fisc="RSSMRA65E25L781T">`
- Si possono usare anche gli apici singoli
  - Ad esempio: `<persona cod_fisc='RSSMRA65E25L781T'>`

## Altro esempio di documento XML

Elenco di libri

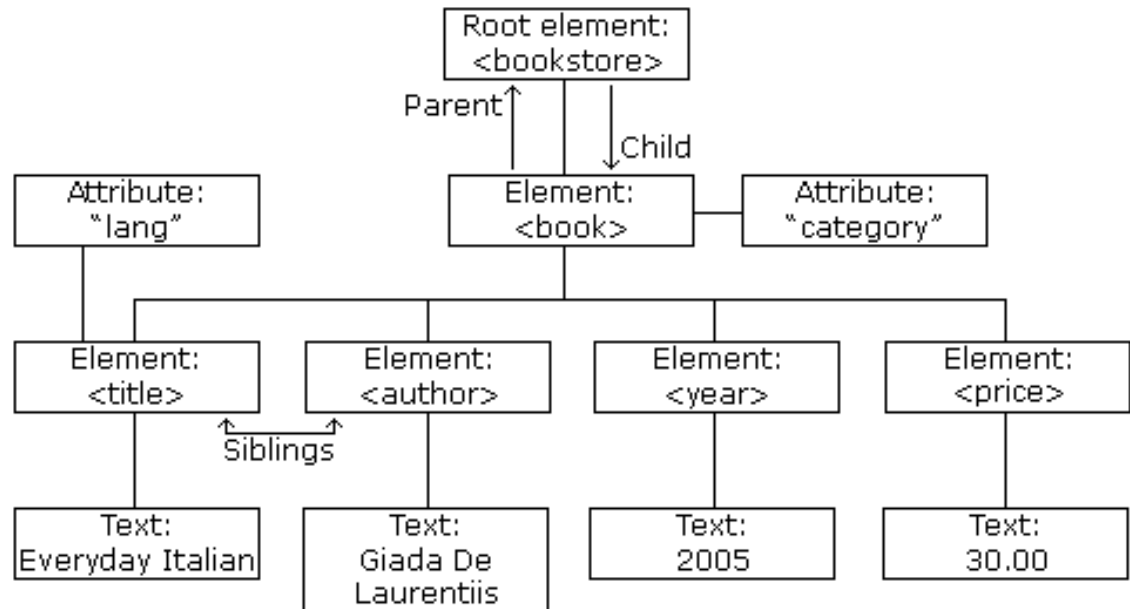
Documento:

**bookstore.xml**

```
<bookstore>
  <book category="COOKING">
    <title lang="en">Everyday
      Italian</title>
    <author>Giada De
      Laurentiis</author>
    <year>2005</year>
    <price>30.00</price>
  </book>
  <book category="CHILDREN">
    <title lang="en">Harry
      Potter</title>
    <author>J K. Rowling</author>
    <year>2005</year>
    <price>29.99</price>
  </book>
  <book category="WEB">
    <title lang="en">Learning
      XML</title>
    <author>Erik T. Ray</author>
    <year>2003</year>
    <price>39.95</price>
  </book>
</bookstore>
```

## Altro esempio di documento XML

Rappresentazione grafica del documento XML **bookstore.xml**



# Elementi vs Attributi

15

## Contenuto rappresentato con attributi

```
<person sex="female">  
  <firstname>Anna</firstname>  
  <lastname>Smith</lastname>  
</person>
```

## Contenuto rappresentato con elementi

```
<person>  
  <sex>female</sex>  
  <firstname>Anna</firstname>  
  <lastname>Smith</lastname>  
</person>
```

# Elementi vs Attributi

16

Nel primo esempio il sesso della persona è stato rappresentato come attributo. Nel secondo come elemento. L'informazione rappresentata è la stessa in entrambe i casi. Non ci sono regole prefissate su quando usare attributi e quando usare elementi per rappresentare informazioni in XML.

Tuttavia se XML viene usato come sintassi per la specifica di informazione **semi-strutturata** è evidente che gli elementi forniscono una rappresentazione più accurata del dato semi-strutturato.



# Elementi vs Attributi

17

## Svantaggi dell'uso di attributi:

- Gli attributi non possono contenere più valori (gli elementi sì)
- Gli attributi non possono contenere informazioni ulteriormente strutturate ad albero (gli elementi sì)
- Gli attributi non possono essere facilmente estesi (gli elementi sì)

Gli attributi sono difficili da leggere e gestire. Meglio usare gli elementi per le informazioni e gli attributi per la meta-informazione. Ad esempio gli identificatori possono essere memorizzati usando attributi.

## Altro esempio di documento XML

Rappresentazione di  
identificatori come  
attributi di elementi

```
<messages>
  <note id="501">
    <to>Tove</to>
    <from>Jani</from>
    <heading>Reminder</heading>
    <body>Don't forget me this
      weekend!</body>
  </note>
  <note id="502">
    <to>Jani</to>
    <from>Tove</from>
    <heading>Re: Reminder</heading>
    <body>I will not</body>
  </note>
</messages>
```

# Sintassi dei **nomi** degli **elementi XML**

19

- Possono essere costituiti da qualsiasi carattere alfanumerico
- Possono includere:
  - Underscore \_
  - Trattino –
  - Punto .
- Possono iniziare solo con lettere, ideogrammi o con il carattere underscore
- Non possono includere:
  - Altri caratteri di punteggiatura
  - Virgolette
  - Apostrofi
  - \$ e %
  - < e >
  - Spazi

# Esempi di nomi XML

20

## Nomi ben formati:

- `<Nome_persona> Maria </Nome_persona>`
- `<Giorno-Mese-Anno> 10/06/2004 </Giorno-Mese-Anno>`
- `<_indirizzo> Via Stella 10 </_indirizzo>`

## Nomi NON ben formati:

- `<Nome persona> Maria </Nome persona>`
- `<Giorno/Mese/Anno> 10/06/2004 </Giorno/Mese/Anno>`
- `<citta'> Verona </citta'>`
- `<1_telefono> 045 1234567 </1_telefono>`
- `<%vendita> 20 </%vendita>`

## Altro esempio di documento XML

Dato a struttura  
variabile

```
<messages>
  <note id="501">
    <to>Tove</to>
    <from>Jani</from>
    <body>Don't forget me this
      weekend!</body>
  </note>
  <note id="502">
    <to>Jani</to>
    <from>Tove</from>
    <heading>Re: Reminder</heading>
    <body>I will not</body>
  </note>
  <note id="505">
    <to>Jani</to><from>Tove</from>
    <body>I will not</body>
    <cc>John</cc>
  </note>
</messages>
```

# Intestazione dei documenti XML

22

Ogni documento XML dovrebbe iniziare con una dichiarazione XML

```
<?xml version="1.0" encoding="US-ASCII" standalone="yes"?>
```

**Versione  
di XML**

**Codifica  
dei  
caratteri**

**Indica se il  
documento è  
conforme ad una  
sintassi esterna  
(standalone="no")**

# Validazione dei documenti XML

Ogni documento XML ben formato rispetta la sintassi base del XML.

Per ogni insieme di documenti XML che debba rappresentare una certa categoria di informazioni è possibile definire un documento XML che ne descrive la sintassi specifica: vale a dire, quali sono i tag ammessi, qual è la struttura dei tag e dei loro attributi, ecc...

La **SINTASSI** di un documento **XML** (file .XML)

SI DESCRIVE ATTRAVERSO

uno **SCHEMA** di documento in **XMLSchema** (file .XSD)

Ogni documento XML è **valido** se è ben formato e rispetta la sintassi specificata nel suo file XSD.