

Introduzione ad XML

MODULO 5

Introduzione ad XML

- eXtensible Markup Language
- Le specifiche sono presenti sul sito <http://www.w3.org/XML>
- Metalinguaggio per la definizione di linguaggi di markup
- Permette di definire la struttura di documenti e dati
- Software: Visual Studio Code, Notepad++, Brackets, ecc

Introduzione ad XML

- XML è un insieme di regole sintattiche per modellare la struttura di documenti e dati:
 - Le regole sono standard. Garantiscono l'indipendenza da una specifica piattaforma hardware e software
 - Non permettono di specificare altre caratteristiche come il tipo o la presentazione dei dati o documenti
- Descrive i dati e non la loro rappresentazione!

A cosa serve XML

- Data Interchange
 - Scambio di informazioni e dati tra più programmi
- Document publishing
 - lo stesso documento XML può essere usato e trasformato per la stampa, il Web, il cellulare, ecc
- Rappresentazione di testi in formato binario (TEI)
 - <https://tei-c.org/>
- Interazione tra database eterogenei
- Android Manifest, interfacce delle activity ...

Perchè XML

- Documenti autodescrittivi
 - La scelta dei nomi può essere fatta per facilitarne la comprensione
- Struttura navigabile dei documenti
 - La struttura ad albero rendono semplice la navigazione
- Platform independence
 - XML è uno standard aperto
- Facile convertibilità
 - La conversione tra formati anche di diversa natura è semplice

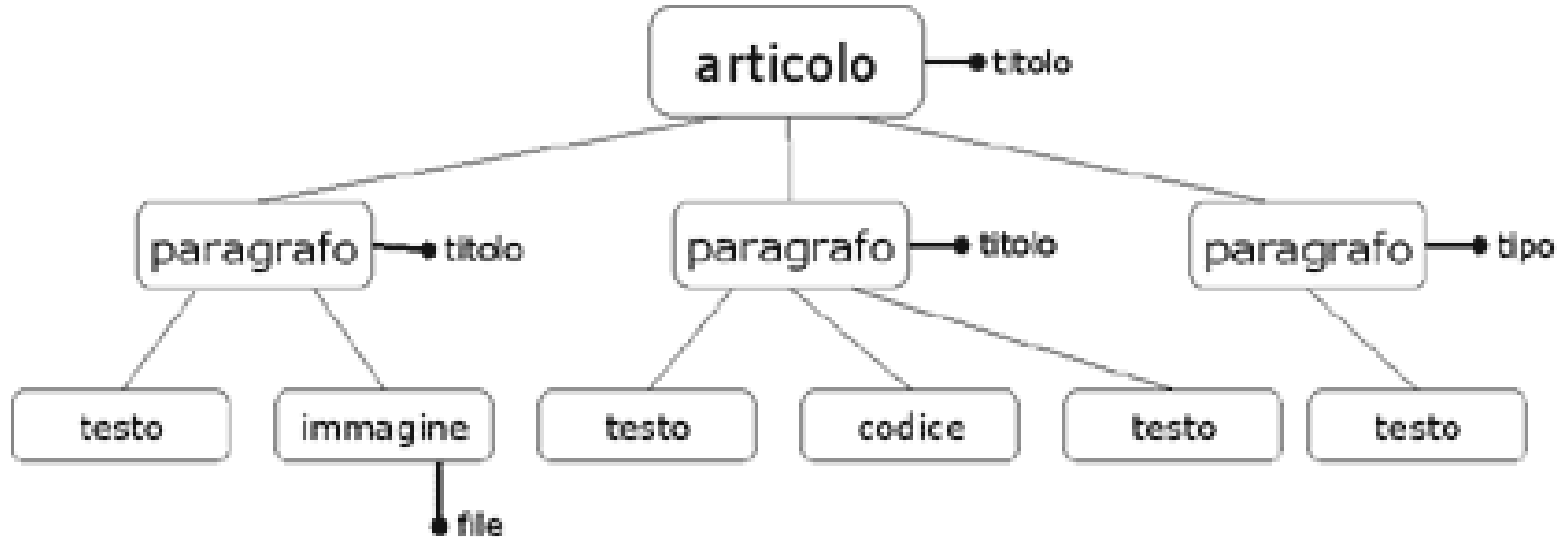
Struttura di un file XML

- Un file XML è un file di testo contenente tag, attributi e testi secondo regole sintattiche ben precise
- Ha un formato aperto e leggibile, simile all'HTML
- Contrariamente all'HTML, però, l'XML è **ESTENSIBILE**
 - Possiamo creare qualsiasi tag e qualsiasi attributi

Struttura di un file XML

- La struttura è gerarchica: albero, detto document tree
- Ogni componente logica viene detta elemento
- Essendo gerarchico, ogni elemento può contenere sottoelementi
- Gli elementi possono avere delle proprietà, dette attributi
- L'elemento principale viene detto ROOT

Esempio di file XML



Esempio di file XML

```
1  <?xml version="1.0" ?>
2  <articolo titolo="Titolo dell'articolo">
3    <paragrafo titolo="Titolo del primo paragrafo">
4      <testo>
5        Blocco di testo del primo paragrafo
6      </testo>
7      <immagine file="immagine1.jpg" />
8    </paragrafo>
9    <paragrafo titolo="Titolo del secondo paragrafo">
10     <testo>
11       Blocco di testo del secondo paragrafo
12     </testo>
13     <codice>
14       Esempio di codice
15     </codice>
16     <testo>
17       Altro blocco di testo
18     </testo>
19   </paragrafo>
20   <paragrafo tipo="bibliografia">
21     <testo>
22       Riferimento ad un articolo
23     </testo>
24   </paragrafo>
25 </articolo>
```

Specifica il tipo di documento e la versione

Root: è il primo elemento

Elemento paragrafo

Sintassi abbreviata

Elemento testo

Attributo

Sintassi standard

Struttura di un file XML

- La prima riga del codice è sempre costituita dalla dichiarazione del tipo di file, dalla versione ed eventualmente dall'encoding:

```
<?xml version="1.0" encoding="UTF-8" ?>
```

- Ogni elemento è costituito da **tag**:

```
<testo>  
| Blocco di testo del secondo paragrafo  
</testo>
```

- Gli attributi si definiscono in questo modo:

```
<paragrafo tipo="bibliografia">
```

Struttura di un file XML

- La struttura deve essere **well-formed**
- Ogni file ha solo **una root**
- XML è **case sensitive**
- I valori degli attributi devono essere sempre tra " o '
- La sintassi per i commenti è
<!-- qui il commento -->

Struttura di un file XML

Esempio di un articolo di giornale

Oggi a Bruxelles vertice della Ue sull'Iraq. Blair scrive agli altri per chiedere una posizione comune

La Nato in difesa della Turchia

Accordo in extremis sull'aiuto all'alleato in prima linea

DAL NOSTRO CORRISPONDENTE FRANCO PAPITTO

BRUXELLES — Tutti hanno atteso per una giornata intera dal piccolo Belgio. Sembrava disponibile la Francia e taceva la Germania; a quel punto Lord Robertson, il segretario generale della Nato, tentava il grande colpo per chiudere finalmente, con una decisione sul sostegno alla Turchia, la settimana più lacerante mai vissuta.

Alle 21 il Belgio aveva strappato quasi tutto ma continuava a ritenere «non soddisfacente» la formulazione del legame fra le decisioni dell'Onu e quelle della Nato. In questo clima di gravi tensioni si riuniscono stamane i ministri degli Esteri dell'Ue nel tentativo di concordare una posizione comune che dovrebbe essere avallata stasera dai capi di governo. Blair ha scritto nei giorni scorsi a tutti i partner europei per chieder loro di non escludere, nel testo che approveranno stasera, un intervento militare contro Saddam. E' una richiesta che metterà sicuramente in grande imbarazzo Schroeder che ha sinora ha detto «no» alla partecipazione del suo paese a qualsiasi intervento, deciso o meno dell'Onu. Aznar, il premier spagnolo, ha chiamato al telefono Chirac che gli avrebbe promesso di fare «tutto il possibile» perché oggi si raggiunga un accordo. Lo stesso Aznar ha scritto al greco Costas Kifitis, presidente di turno dell'Ue, per chiedergli che la dichiarazione europea di stasera solleciti una «rigida applicazione» della risoluzione 1441 dell'Onu da parte di Saddam.

Bruxelles ha presentato tre emendamenti sulla solidarietà al governo di Ankara

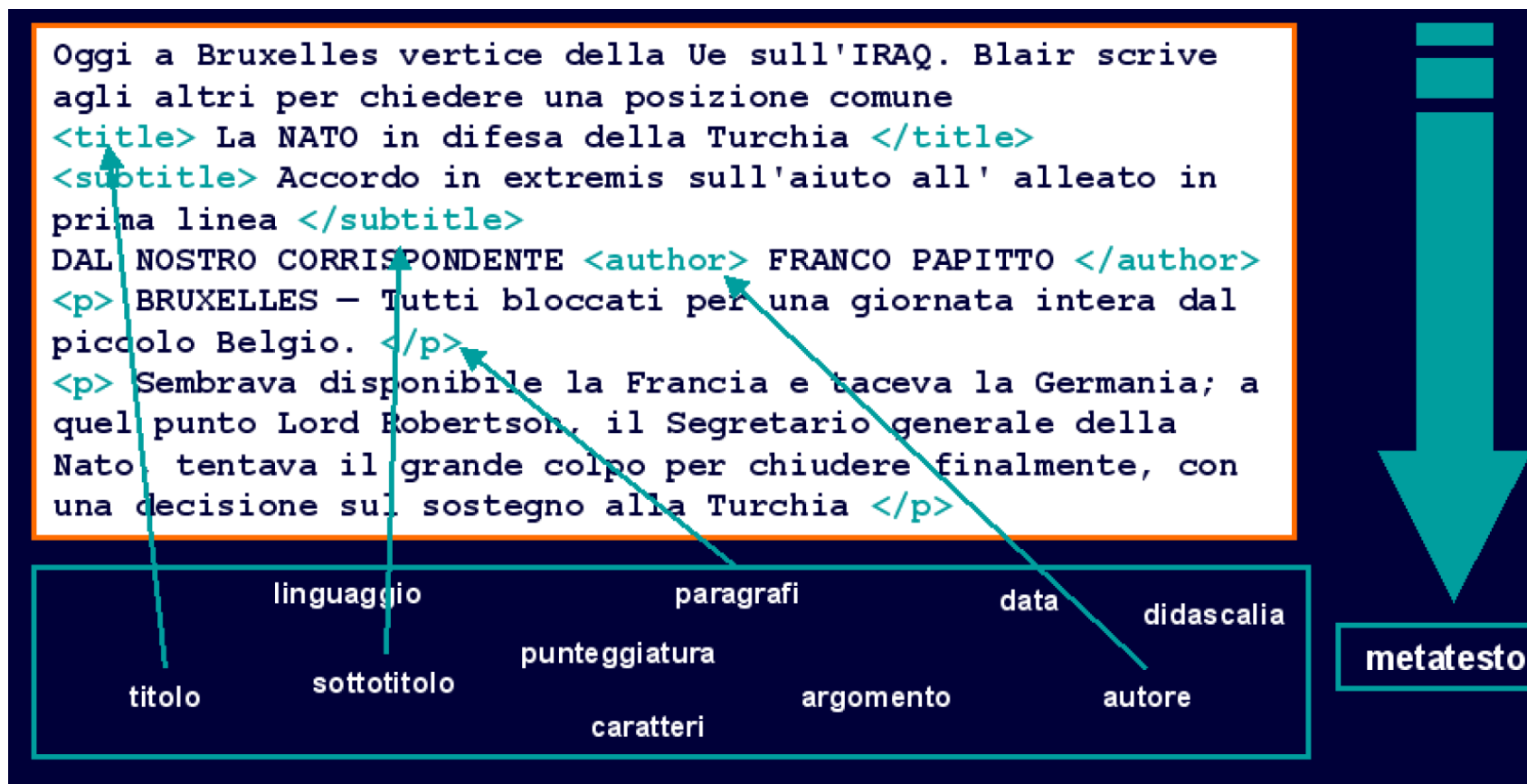
Il segretario generale della Nato George Robertson

NUM

titolo
sottotitolo
linguaggio
punteggiatura
caratteri
paragrafi
argomento
data
di dascalia
autore
metatesto

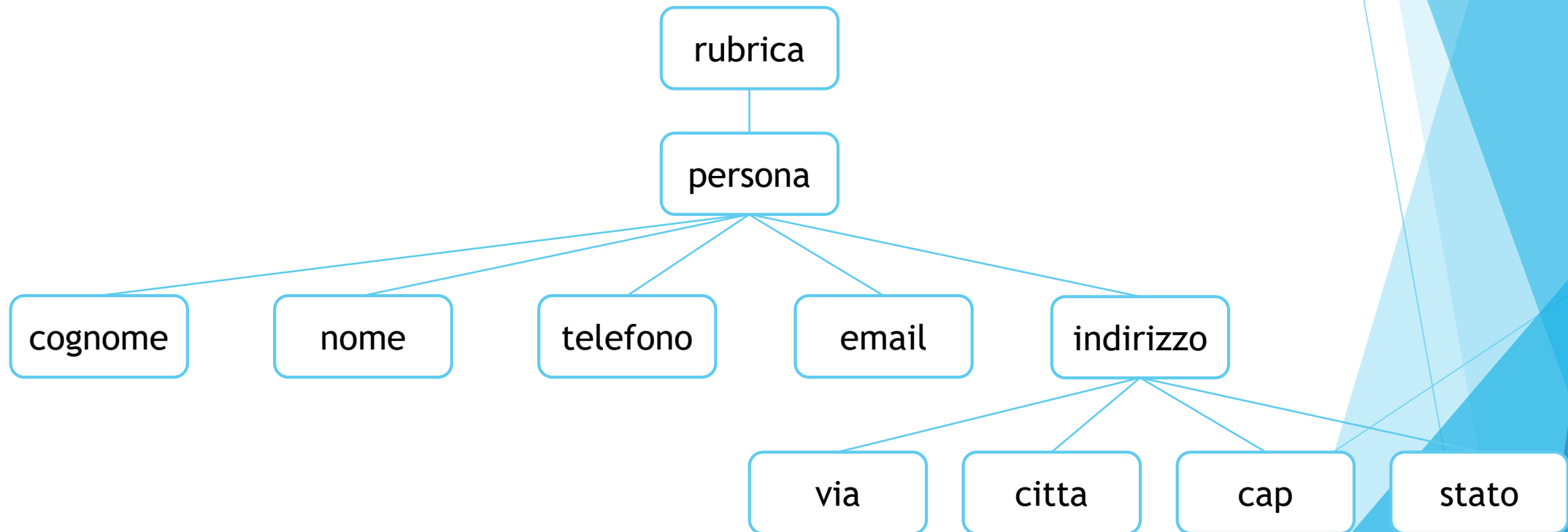
Struttura di un file XML

Esempio di un articolo di giornale: testo digitale



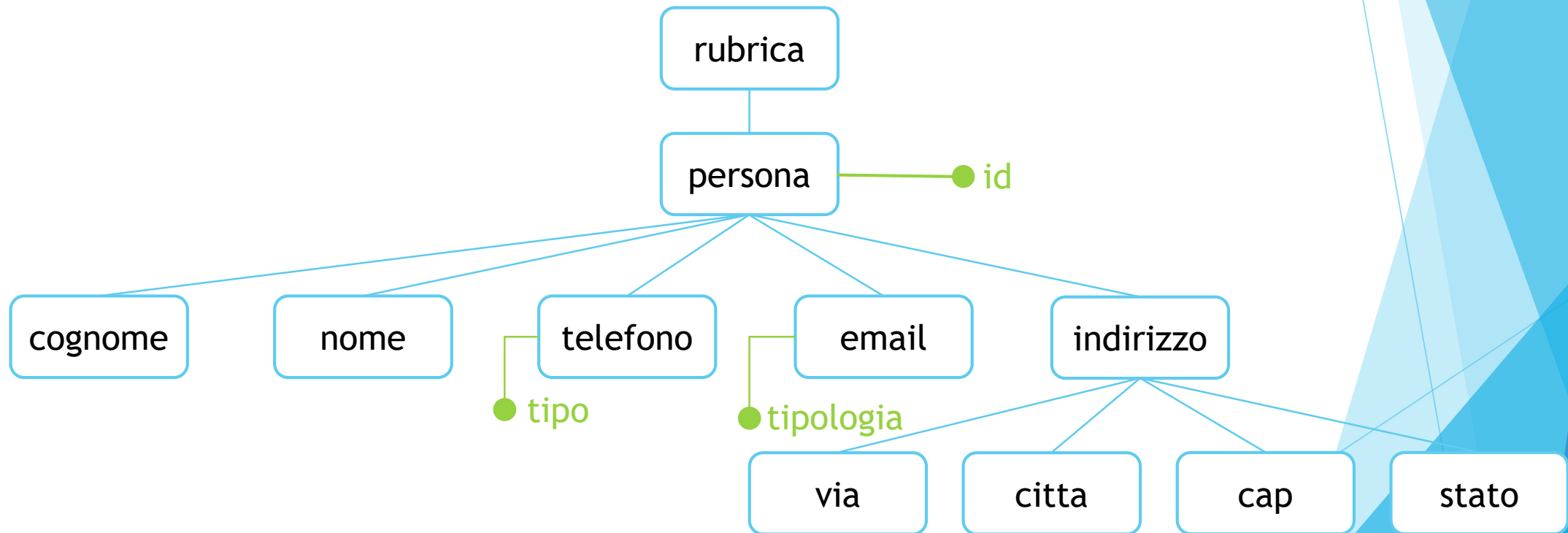
Struttura di un file XML

Esercizio: creare il file *rubrica1.xml* relativo al seguente albero inserendo dei dati fittizi



Struttura di un file XML

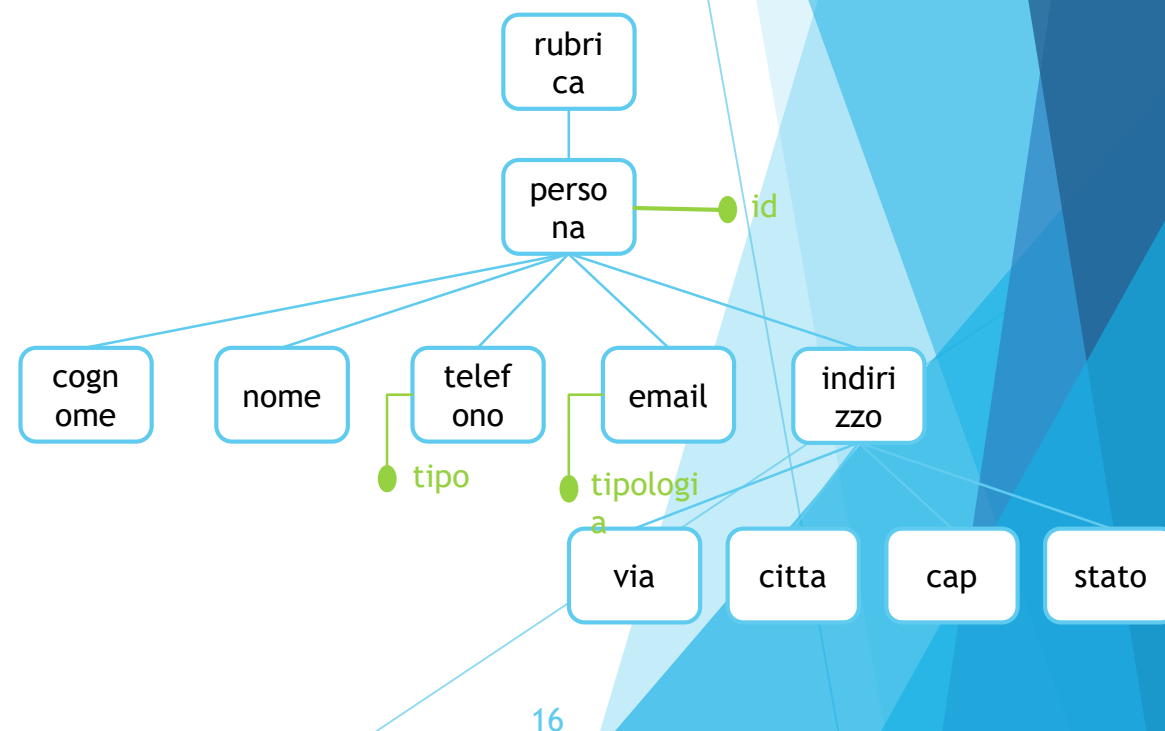
Esercizio: modificare il file *rubrica1.xml* con i dati sotto e salvare il nuovo file con nome *rubrica.xml*



Struttura di un file XML

Esercizio: creare una copia del file *rubrica.xml*, inserire i dati sotto e salvarlo con nome *rubrica2.xml*

persona: id	0001	0002
cognome	Rossi	Bianchi
nome	Paolo	Maria
telefono tipo	123 456 789 cellulare 525 652 235 ufficio	987 654 321 ufficio <+39> 085 252 536 fisso
email tipologia email	paolo.rossi@gmail.com privato ufficiopaolo.rossi@libero .it ufficio	mariabianchi@tin.it ufficio
via	Tal dei tali 45	D'Ascanio, 75
citta	L'Aquila	Pescara
cap	60000	65127
stato	Italia	Italia



Struttura di un file XML

- Come per l'HTML, anche XML prevede l'uso degli oggetti speciali, per rappresentare correttamente alcuni caratteri

Entità	Carattere corrispondente
&	&
<	<
>	>
"	"
'	'

Struttura di un file XML

Esercizio: dato il file xml determinare l'albero corrispondente

```
<?xml version="1.0"?>
```

```
<BOOKLIST>
```

```
<BOOK>
```

```
<TITLE edition="2000">The XML Companion</TITLE>
```

```
<AUTHOR>Neil Bradley</AUTHOR>
```

```
</BOOK>
```

```
...
```

```
<BOOK>
```

```
<TITLE edition="2000" type="XML">
```

```
Data on the Web
```

```
</TITLE>
```

```
<AUTHOR>Serge Abiteboul</AUTHOR>
```

```
<AUTHOR>Peter Buneman</AUTHOR>
```

```
<AUTHOR>Dan Suciu</AUTHOR>
```

```
</BOOK>
```

```
</BOOKLIST>
```

BOOKS.xml

XML e grammatica

- XML offre la possibilità di definire i tag a seconda delle necessità, ma per evitare confusione è necessario un meccanismo che ne vincoli l'utilizzo all'interno dei documenti: grammatica.
- La **grammatica** è un insieme di regole che indica quali vocaboli possono essere utilizzati e con che struttura è possibile comporre frasi.
- Un documento può essere valido rispetto ad una grammatica, ma non rispetto ad un'altra!

XML Schema

- Come si costruiscono le grammatiche?
 - DTD - Document Type Definition
 - XML Schema
- Entrambi forniscono la descrizione formale di una grammatica per XML.

XML Schema

- XML Schema utilizza la sintassi XML per definire la grammatica.
- La struttura generale è la seguente:

```
1  <?xml version="1.0"?>
2  <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
3  ...
4  </xs:schema>
```

- Qual è l'elemento root?
- Che rappresenta la xs:?

XML Schema

- Come vengono definiti gli elementi?

```
<xs:element name="note">
```

- Come specifichiamo il tipo di dato da inserire?

```
<xs:element name="to" type="xs:string"/>
```

- In XML esistono tipi di dati semplici e complessi

XML Schema: tipi di dato

- In XML esistono due categorie di tipi di dati:
 - Semplici (sotto la lista dei più usati)

Codifica XML Schema	Descrizione
xs:string	Stringa di caratteri
xs:integer	Numero intero
xs:decimal	Numero decimale
xs:boolean	Valore booleano: vero o falso
xs:date	Data
xs:time	Orario
xs:uriReference	Inserimento URL

- **Complessi**

XML Schema: tipi di dato semplici

- XML Schema permette di definire tipi di dati semplici, ma personalizzati.
- Ad esempio, possiamo definire l'età di una persona come:

```
<xs:element name="eta" >
  <xs:simpleType>
    <xs:restriction base="xs:integer">
      <xs:minInclusive value="1" />
      <xs:maxInclusive value="130" />
    </xs:restriction>
  </xs:simpleType>
</xs:element>
```

XML Schema: tipi di dato complessi

- I tipi di dato complessi si riferiscono ad elementi che possono contenere altri elementi e possono avere attributi.
- Definire un elemento di tipo complesso corrisponde a definire la relativa struttura.

XML Schema: tipi di dato complessi

- Lo schema generale per la definizione di un elemento di tipo complesso è il seguente:

```
<xs:element name="NOME_ELEMENTO">
  <xs:complexType>
    ... Definizione del tipo complesso ...
    ... Definizione degli attributi ...
  </xs:complexType>
</xs:element>
```

XML Schema: tipi di dato complessi

- In XML Schema i costruttori di tipi complessi previsti sono:

Costruttori	Descrizione
<xs:sequence>	Consente di definire una sequenza ordinata di sottoelementi
<xs:choice>	Consente di definire un elenco di sottoelementi alternativi
<xs:all>	Consente di definire una sequenza non ordinata di sottoelementi

- Per ciascuno di questi costruttori e per ciascun elemento è possibile definire il numero di occorrenze previste utilizzando gli attributi **minOccurs** e **maxOccurs**.

XML Schema: tipi di dato complessi

- Esercizio: dato il seguente file *note.xml* creare l'XML Schema corrispondente. Salvare il file con nome *note.xsd*

```
1  <?xml version="1.0"?>
2  <note>
3      <to>Tove</to>
4      <from>Jani</from>
5      <heading>Reminder</heading>
6      <body>Don't forget me this weekend!</body>
7  </note>
```

XML Schema: tipi di dato complessi

- In XML Schema gli attributi vengono considerati come un tipo di dato complesso:

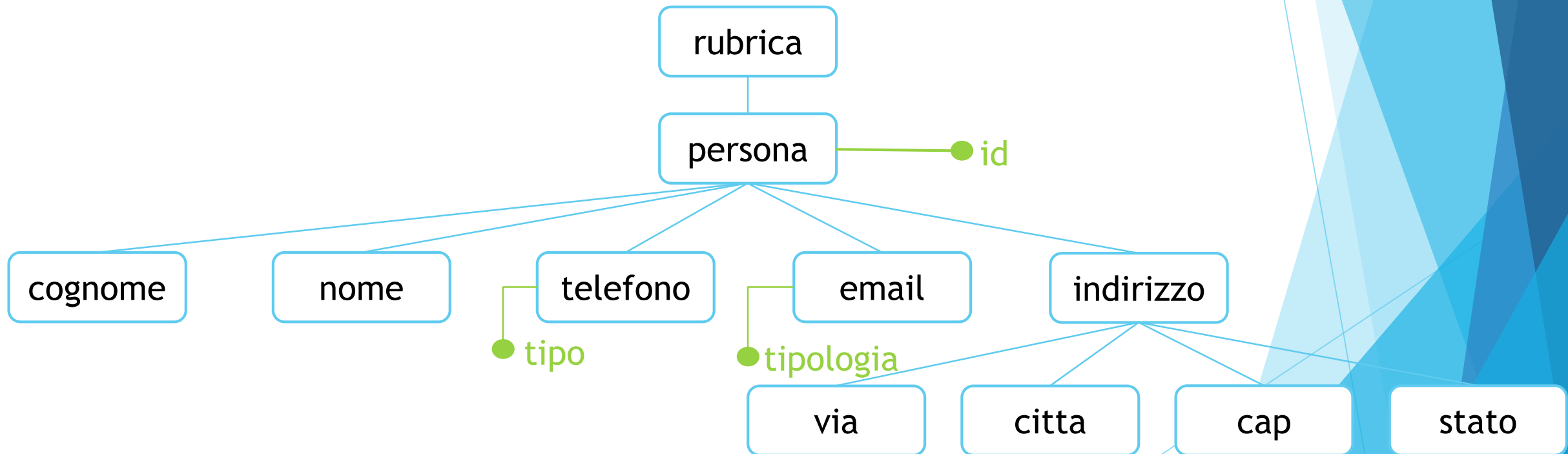
```
<xs:attribute name="titolo" type="xs:string" use="required" />
```

- L'attributo **use** permette di indicare se l'attributo è obbligatorio (required) o se ha un valore predefinito (default). In quest'ultimo caso occorre inserire anche l'attributo **value**

```
<xs:attribute name="titolo" type="xs:string" use="default" value="test" />
```

XML Schema: tipi di dato complessi

- Esercizio: dato il file *rubrica.xml* creare l'XML Schema corrispondente. Salvare il file con nome *rubrica.xsd*



XML Schema: combinazione di grammatiche

- Una delle caratteristiche principali dell'XML Schema è la possibilità di integrare elementi derivanti da grammatiche diverse.
- Questa caratteristica consente di riutilizzare parti di grammatiche già definite evitando di dover rifare parte di lavoro già fatto in altri ambiti.
- La composizione di linguaggi pone almeno due tipi di problemi:
 - la validazione: a quale schema si deve fare riferimento per validare un documento XML "ibrido"?
 - due linguaggi potrebbero avere tag ed attributi con lo stesso nome

XML: Namespace

- Un **namespace** è un insieme di nomi di elementi e nomi di attributi identificati univocamente da un identificatore.
- L'identificatore univoco individua l'insieme dei nomi distinguendoli da eventuali omonimie in altri namespace.
- In un documento XML si fa riferimento ad un namespace utilizzando un attributo speciale (**xmlns**) associato al root element

```
<note xmlns="http://www.w3schools.com/xml/note" >
```

XML: Namespace

- Per mettere in relazione un namespace con il relativo XML Schema occorre dichiararlo nel root element come nel seguente esempio:

```
1 <?xml version="1.0"?>
2
3 <note xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4       xmlns="http://www.dominio.it/xml/note"
5       xmlns="http://www.dominio.it/xml/rubrica"
6       xsi:schemaLocation="http://www.dominio.it/xml/note note.xsd"
7       xsi:schemaLocation="http://www.dominio.it/xml/rubrica rubrica.xsd">
```

Attenzione!!!!

XML: Namespace

- Per mettere in relazione un namespace con il relativo XML Schema occorre dichiararlo nel root element come nel seguente esempio:

```
<nt:note xmlns:xsi="http://www.w3.org/2003/XMLSchema"
          xmlns:rb="http://www.miosito.it/XMLSchema/rubrica"
          xmlns:nt="http://www.miosito.it/XMLSchema/note"
          nt:noNamespaceSchemaLocation="note.xsd"
          rb:noNamespaceSchemaLocation="rubrica.xsd">
```

Attenzione!!!!

XML: Namespace

- È possibile combinare più namespace facendo in modo che ciascun elemento utilizzato faccia riferimento al proprio namespace.
- Occorre tener presente che quando si fa riferimento ad un namespace, questo riferimento vale per l'elemento corrente e per tutti gli elementi contenuti, a meno che non venga specificato un diverso namespace.

XML: Namespace

- Riportare il riferimento ad un namespace per ogni elemento è di solito scomodo e rende di difficile lettura il documento XML.
- È possibile creare delle abbreviazioni per fare riferimento ai namespace.
- Queste abbreviazioni sono costituite da caratteri alfanumerici seguiti da due punti (:) dichiarati nel root element ed utilizzati come prefissi dei nomi degli elementi.

XML: Namespace

- Esercizio: creare un sistema di gestione delle note più complesso in cui gli elementi **to** e **from** sono elementi di **rubrica**. Salvare il file con nome *note_complesse.xml*
- Istruzioni:
 - combinare i file `note.xml` e `rubrica.xml`;
 - inserire correttamente i namespace
 - creare una nota tra la persona:
 - (from) Paolo Rossi
 - (to) Maria Bianchi
 - (heading) "Appuntamento"
 - (body) "<Data: 16/05/2017 Ore 08:30> Richiesta informazioni"

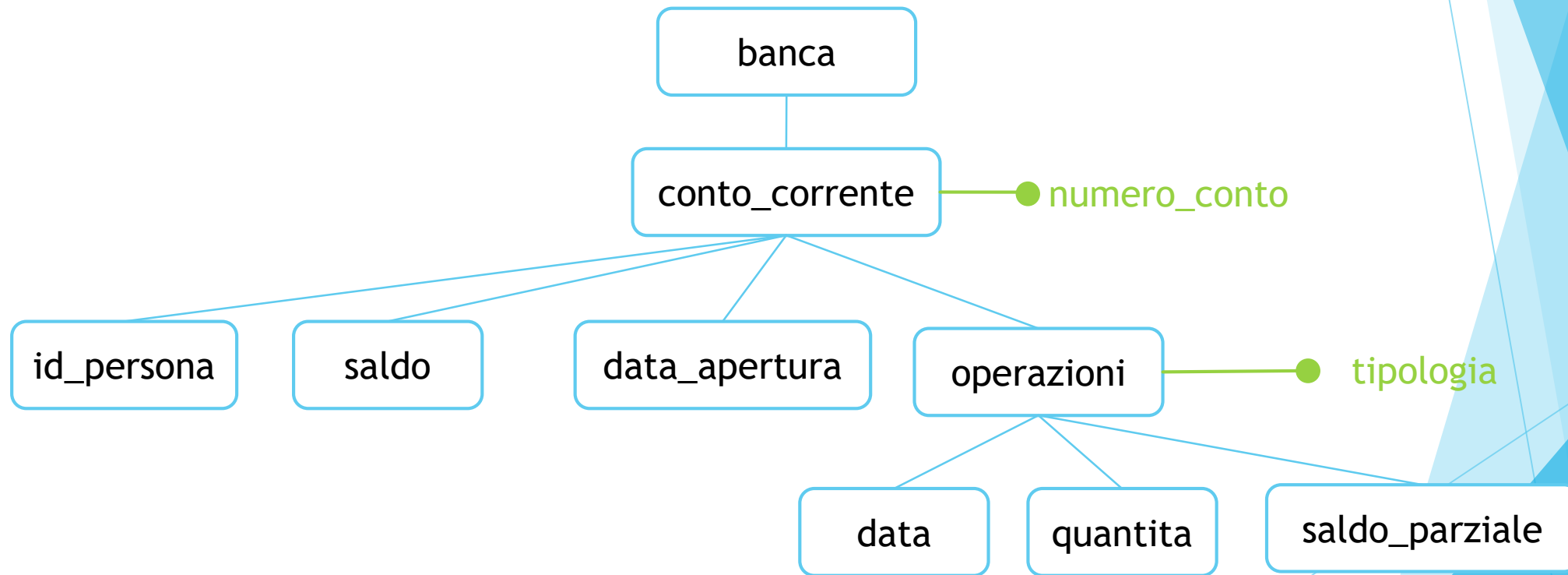
XML Schema: integrazione della grammatica

- La sintassi per integrare la grammatica al file XML è la seguente:

```
1 <?xml version="1.0"?>
2
3 <note xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
4     xsi:noNamespaceSchemaLocation="note.xsd">
5     <to>Tove</to>
6     <from>Jani</from>
7     <heading>Reminder</heading>
8     <body>Don't forget me this weekend!</body>
9 </note>
```

Conto Corrente:

- **Esercizio:**
 - Creare il file XML conto_corrente.xml come da albero



Conto Corrente:

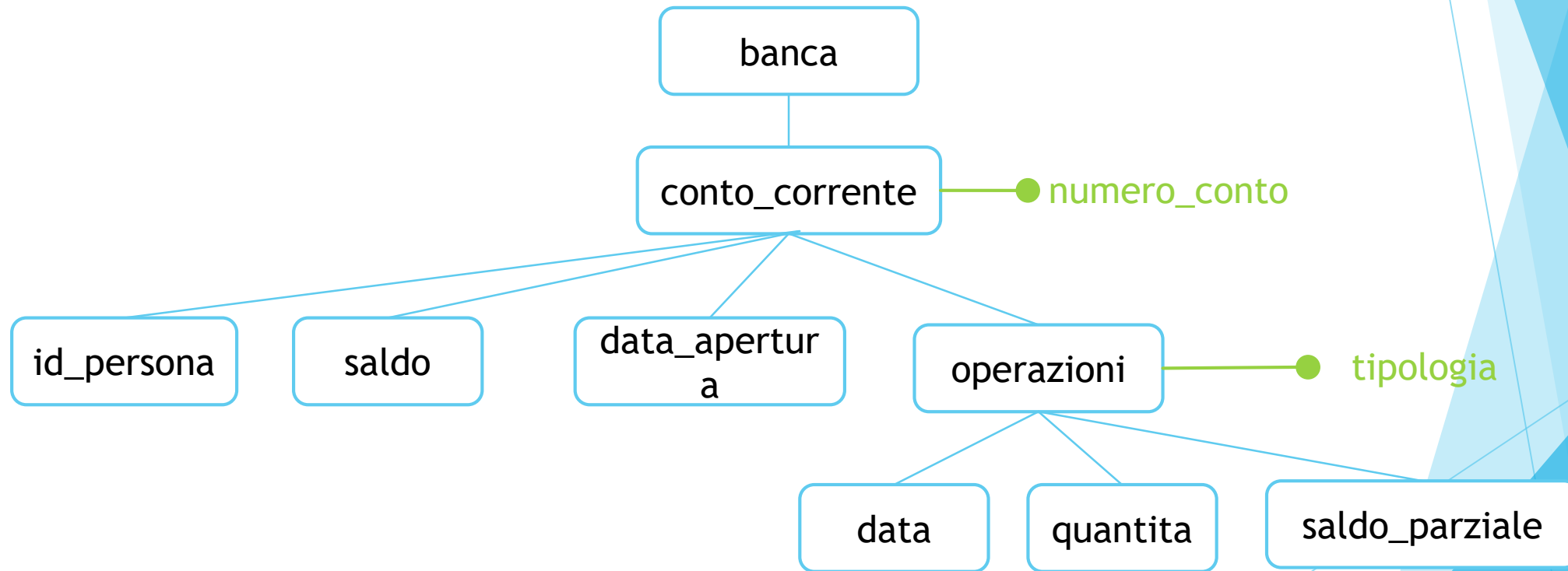
Creare il file XML conto_corrente.xml come da albero

numero_conto		132524	
id_persona		001	
saldo		250	
data_apertura		07/09/2017	
operazioni			
tipologia	data	quantita	saldo_parziale
versamento	07/09/2017	250	250

numero_conto		132523	
id_persona		002	
saldo		700	
data_apertura		10/12/2016	
operazioni			
tipologia	data	quantita	saldo_parziale
versamento	10/12/2016	100	100
prelievo	31/01/2017	200	-100
versamento	15/04/2017	400	300
versamento	07/09/2017	600	900
prelievo	08/09/2017	200	700

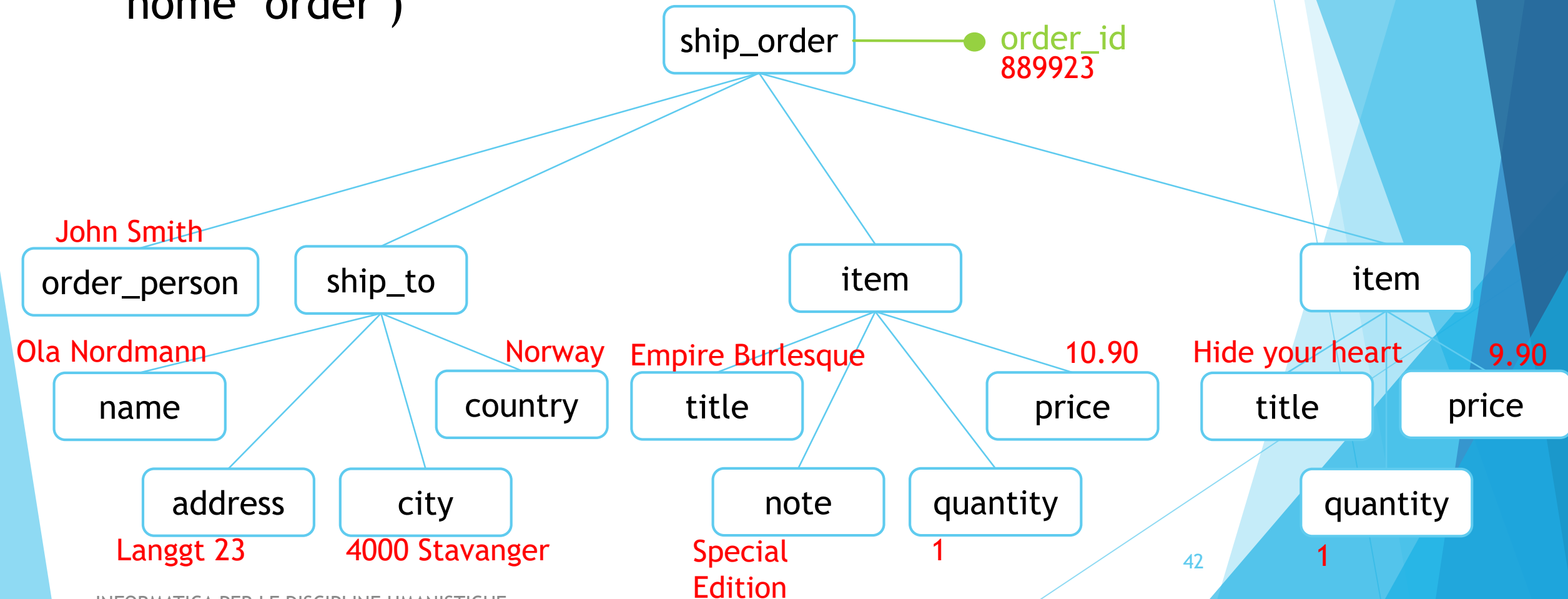
Conto Corrente:

- Esercizio: creare il file xsd del seguente albero (salvare con nome conto_corrente.xsd)



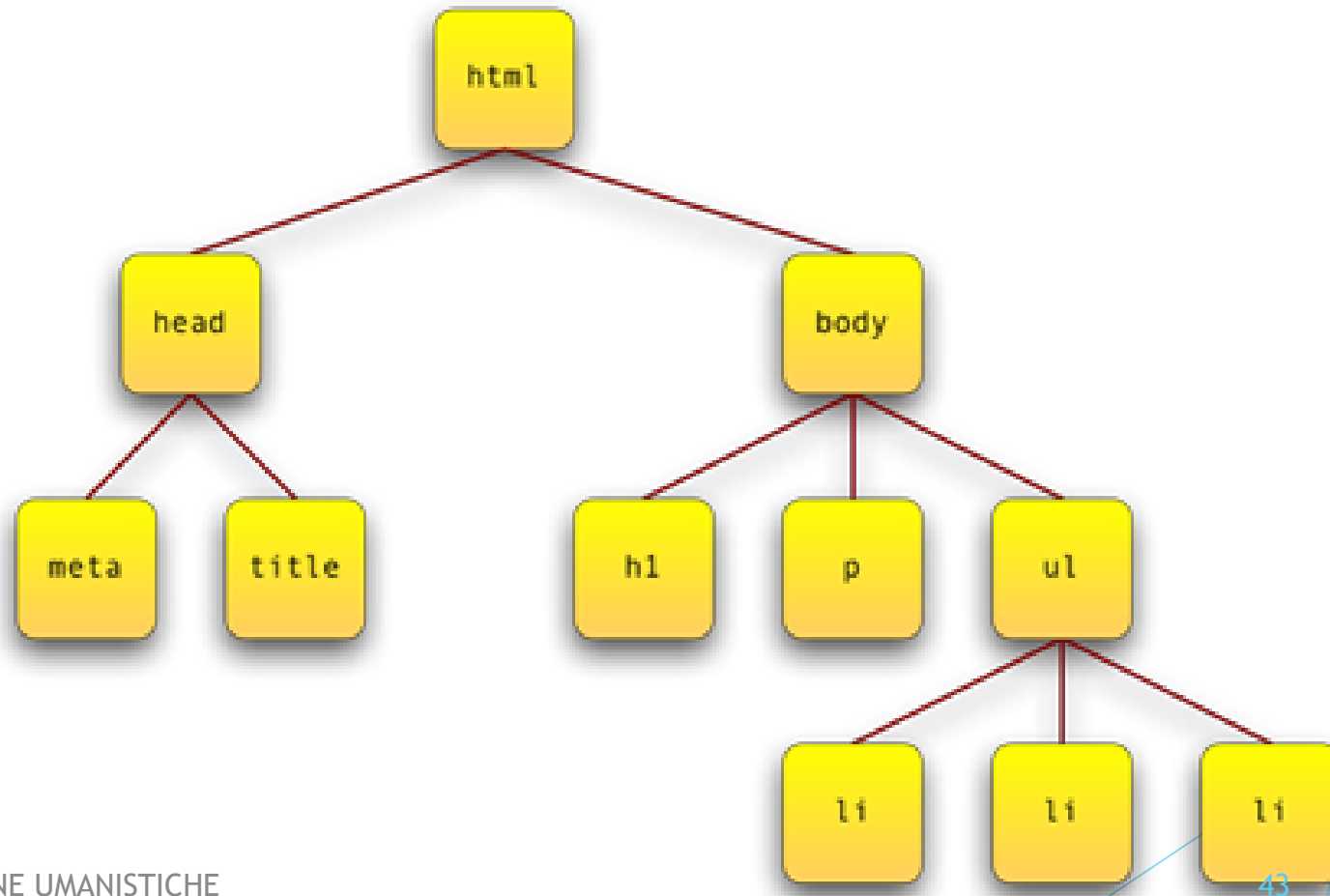
XML Schema: ancora esercizi

- Esercizio: creare il file xsd e xml del seguente albero (salvare con nome "order")



XML Schema: ancora esercizi

- Esercizio: creare il file xsd e xml del seguente albero (salvare con nome html)



XML: esercizio riepilogativo

- Creare l'albero che descrive un curriculum europeo, includendo dati personali, esperienze formative ed esperienze professionali
- Creare l'XML Schema associato all'albero
- Creare il file XML, inserendo le proprie informazioni personali
- Verificare che il file XML sia valido rispetto allo XML Schema

XML: esercizio riepilogativo

- Creare l'albero che descrive il piano di studio di uno studente per il corso di laurea triennale
- Creare l'XML Schema associato all'albero
- Creare il file XML, inserendo le proprie informazioni personali
- Verificare che il file XML sia valido rispetto allo XML Schema

JSON: JavaScript Object Notation

JSON

- ▶ JavaScript Object Notation
- ▶ Formato di interscambio
- ▶ Adatto alle comunicazioni tra server e client
- ▶ Molto leggero rispetto a XML
- ▶ Nato nel mondo Web con l'utilizzo di AJAX

JSON

- ▶ Subset di tipi:
 - ▶ Booleani
 - ▶ Stringhe (si usano le virgolette)
 - ▶ Numeri (interi, decimali, virgola mobile)
 - ▶ Array
 - ▶ Array associativi
 - ▶ Null

Confronto XML - JSON

```
<?xml version="1.0" ?>
<rubrica>
  <persona id="0123">
    <cognome>Di Nardo Di Maio</cognome>
    <nome>Simone</nome>
    <telefono tipo="cellulare">0039 123 45 67 890</telefono>
    <email tipologia="privata">simone.dinardo@unich.it</email>
    <indirizzo>
      <via>Via Tal dei Tali, 71</via>
      <citta>Pescara</citta>
      <cap>65127</cap>
      <stato>Italia</stato>
    </indirizzo>
  </persona>
</rubrica>
```

Confronto XML - JSON

```
{
  [
    {
      "id": "0123",
      "cognome": "Di Nardo Di Maio",
      "nome": "Simone",
      "telefono": {
        "tipo": "cellulare",
        "numero": "0039 123 45 67 890"
      },
      "email": {
        "tipologia": "privata",
        "indirizzo": "simone.dinardo@unich.it"
      },
      "indirizzo": {
        "via": "Via Tal dei Tali, 71",
        "citta": "Pescara",
        "cap": "65127",
        "stato": "Italia"
      }
    }
  ]
}
```

INFORMATICA PER LE DISCIPLINE UMANISTICHE

JSON: Esercizio

- ▶ Riprendere l'esercizio di XML sul CV
- ▶ Trasformare l'XML in JSON