

XML-based languages

XML

- Extensible markup language - Μία απλή έκδοση της sgml
- Σχεδιασμένη για να εισάγει την sgml στο διαδίκτυο
- Μια μετα-γλώσσα (meta-language)
 - Μια γλώσσα που παράγει γλώσσες
- Θα αντικαταστήσει σταδιακά την HTML
- **XHTML** μία αναδιαμόρφωση της HTML 4 στην XML 1.0

Αναμενόμενα κέρδη

- Αποθήκευση μια φορά και μορφοποίηση πολλές φορές
- Ανεξαρτησία υλικού - λογισμικού
- Συγκέντρωση δεδομένων μια φορά και ανταλλαγή πολλές φορές
- Ταχύτερη εστιασμένη αναζήτηση
- Μικρότερη συμφόρηση του δικτύου
- Οι μηχανές αναζήτησης αναζητούν συγκεκριμένες ετικέτες (tags) στον κώδικα XML
 - Ταχύτερα
 - Με μεγαλύτερη ακρίβεια

XML

HTML

- Χρησιμοποιεί tags ανάμεσα στο κείμενο...
 - ...για να περιγράψει το layout της σελίδας
`<p> Alan, 42 years, <i>agb@abc.com</i>`
- Σχεδιάστηκε ειδικά για να περιγράψει την παρουσίαση και όχι το περιεχόμενο

XML

- Σχεδιάστηκε ειδικά για να περιγράψει το περιεχόμενο (content) και όχι την παρουσίαση μιας σελίδας
- Βασικές διαφορές από την HTML
 - Μπορεί κανείς να ορίσει νέα tags κατά βούληση
 - Nested tags σε οποιοδήποτε βάθος
 - Ένα έγγραφο XML μπορεί προαιρετικά να περιέχει μια περιγραφή της γραμματικής του
- Τα tags δομούν το περιεχόμενο
`<person><name>Kostas</name>...</person>`
 - Το πως θα εμφανιστούν ορίζεται ξεχωριστά από κάποιο stylesheet (XSL)

XML

- **Ο ρόλος της XML**
 - Προτάθηκε σαν μια markup γλώσσα περιγραφής εγγράφων
 - Καταγωγή απο την *SGML* (ψηφιακές βιβλιοθήκες)
 - Εξελεύσεται όμως σε ένα παγκόσμιο πρότυπο για ανταλλαγή πληροφορίας
- **Βασικό συστατικό της XML είναι το element**
 - Τιμή του element → Κείμενο που περικλείεται από ένα ζεύγος tags
 - Εκτός από την λογική δομή (elements), τα tags περιγράφουν και την φυσική δομή (entities)

```
<person>  
  <name>Alan</name>  
  <age>42</age>  
  <email>alan@abc.com</email>  
  <email>abrown@mail.com</email>  
</person>
```

example

Book Catalogue

id

author

title

price

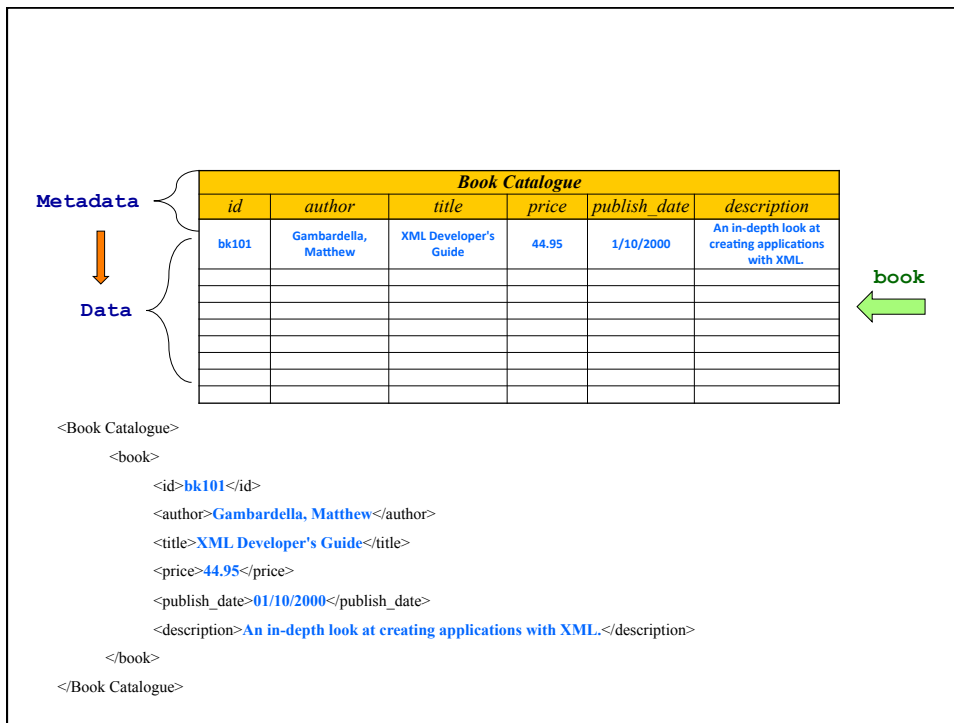
publish_date

description

Τι κάνουμε εδώ;

Excel, DB, ή κάτι... άλλο;

Πλεονεκτήματα / Μειονεκτήματα;



Book Catalogue					
<i>book_id (format)</i>	<i>author (language)</i>	<i>title (language)</i>	<i>price (currency)</i>	<i>publish_date (calendar type)</i>	<i>description (language)</i>
bk101	Gambardella, Matthew	XML Developer's Guide	44.95	01/10/2000	An in-depth look at creating applications with XML.
817525766-0	Αναγνωστόπουλος, Ιωάννης	Τεχνολογίες Διαδικτύου	88.00	01/03/2009	Βασικές αρχές τεχνολογιών Διαδικτύου και προγραμματισμού.

(More Metadata inside the parentheses)
οπότε πως είναι ο κώδικάς μου...;

Πρέπει να περιγράψουμε τα data που περιγράφουν
τα metadata
(meta-metadata;)

Π.χ. το book_id format ενός βιβλίου είναι το ISBN

Για να δούμε πως γράφεται σε XML αυτό...

```
<id format="ISBN">817525766-0</id>
```

Book Catalogue					
<i>id (format)</i>	<i>author (language)</i>	<i>title (language)</i>	<i>price (currency)</i>	<i>publish_date (calendar type)</i>	<i>description (language)</i>
bk101	Gambardella, Matthew	XML Developer's Guide	44.95	01/10/2000	An in-depth look at creating applications with XML.
817525766-0	Αναγνωστόπουλος, Ιωάννης	Τεχνολογίες Διαδικτύου	88.00	01/03/2009	Βασικές αρχές τεχνολογιών Διαδικτύου και προγραμματισμού.

```

      data
      / \
     /   \
    /       \
   /         \
  /           \
 /             \
<id format="ISBN">817525766-0</id>
  /         \
 /           \
/             \
Metadata     Metadata

```

id: element

format: attribute_of_element(id)

ISBN: value_of_attribute(format)

817525766-0: value_of_element(id)

```

<Book Catalogue>
  <book>
    <id>bk101</id>
    <author language="English">Gambardella, Matthew</author>
    <title language="English">XML Developer's Guide</title>
    <price currency="USD">44.95</price>
    <publish_date calendar_type="Gregorian Little Endian">01/10/2000</publish_date>
    <description language="English">An in-depth look at creating applications with XML. </description>
  </book>
  <book>
    <id format="ISBN">817525766-0</id>
    <author language="Greek">Αναγνωστόπουλος, Ιωάννης</author>
    <title language="Greek">Τεχνολογίες Διαδικτύου</title>
    <price currency="Euro">88.00</price>
    <publish_date calendar_type="Gregorian Little Endian">01/03/2009</publish_date>
    <description language="Greek">Βασικές αρχές τεχνολογιών Διαδικτύου και προγραμματισμού.</description>
  </book>
</Book Catalogue>

```

XML

Elements, Attributes

```

<root>
  <child>
    <subchild>.....</subchild>
  </child>
</root>

```

```

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

```

Elements

person, firstname, lastname

Attribute

sex (value = "female")

- Οι ιδιότητες δεν μπορούν να περιέχουν πολλαπλές τιμές (τα child elements μπορούν)
- Οι ιδιότητες δεν πρέπει να αλλάζουν συχνά
- Οι ιδιότητες δεν μπορούν να περιγράψουν δομή (τα child elements μπορούν)

XML

XML attributes

- Ένα **element** μπορεί να έχει κανένα, ένα ή πολλά **attributes**
 - Περιγράφουν ιδιότητες (properties) του **element**

```
<product>
  <name language="French">trompette no 6</name>
  <price currency="Euro"> 420.12 </price>
  <address format="XL1245" language="French">
    <street>31 rue Croix-Bosset</street>
    <zip>92874</zip>
  </address>
</product>
```

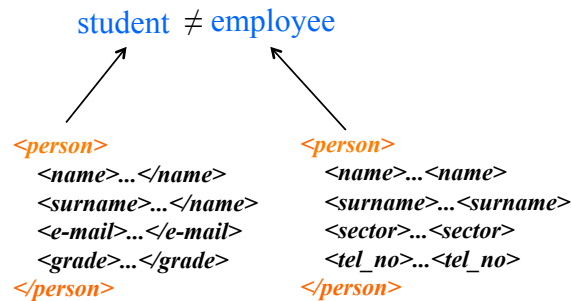
- Όπως και τα **elements**, τα **attributes** ορίζονται από τον χρήστη

XML Namespaces

- Τα ονόματα των **elements** ενός αρχείου χαρακτηρίζονται από μια ετικέτα (super label)
- Το label αυτό ονομάζεται **namespace** και αποτελεί το όνομα της **συλλογής** των ονομάτων του αρχείου
- Το όνομα ενός namespace παίρνει τη μορφή ενός URI, το δεν έχει **link πουθενά!** Χρησιμοποιείται αυτή η αναπαράσταση γιατί είναι **μοναδική**
- συνήθως μοιάζει με ένα URL (διεύθυνση Διαδικτυακού Τόπου)

XML

Παράδειγμα name conflict



XML

Name Conflicts

Εφόσον τα ονόματα των element στην XML δεν είναι προκαθορισμένα (δεσμευμένα), ενδέχεται δύο διαφορετικά XML documents να χρησιμοποιούν το ίδιο element name

Λύση ... τα XML Namespaces

Το XML namespace attribute τοποθετείται στο αρχικό tag ενός element και έχει την παρακάτω σύνταξη:

`xmlns:namespace-prefix="namespaceURI"` ή

`xmlns=namespaceURI` (obsolete)

Όταν ένα namespace καθορίζεται στο αρχικό tag ενός element, όλα τα child elements με το ίδιο namespace-prefix συνδέονται με το ίδιο namespace

XML

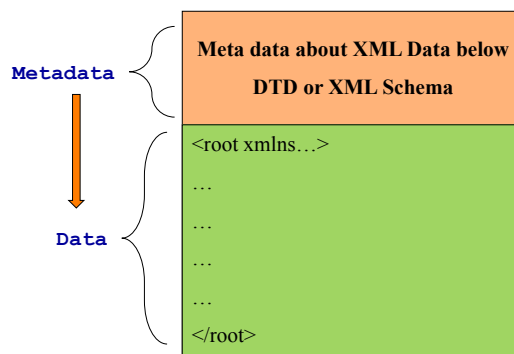
Λύση του name conflict με χρήση Namespace

```
<person xmlns:janag="http://www.anagnostopoulos.name/my_xmlns#" >  
  <janag:name>...</janag:name>  
  <janag:surname>...</janag:surname>  
  <janag:e-mail>...</janag:e-mail>  
  <janag:grade>...</janag:grade>  
</person>
```

```
<person xmlns:p="http://gr.linkedin.com/pub/agissilaos-papantoniou/4/209/758/agis_xmlns#">  
  <p:name>...</p:name>  
  <p:surname>...</p:surname>  
  <p:sector>...</p:sector>  
  <p:tel_no>...</p:tel_no>  
</person>
```

XML

Και τέλος λίγα λόγια για τα metadata των XML data

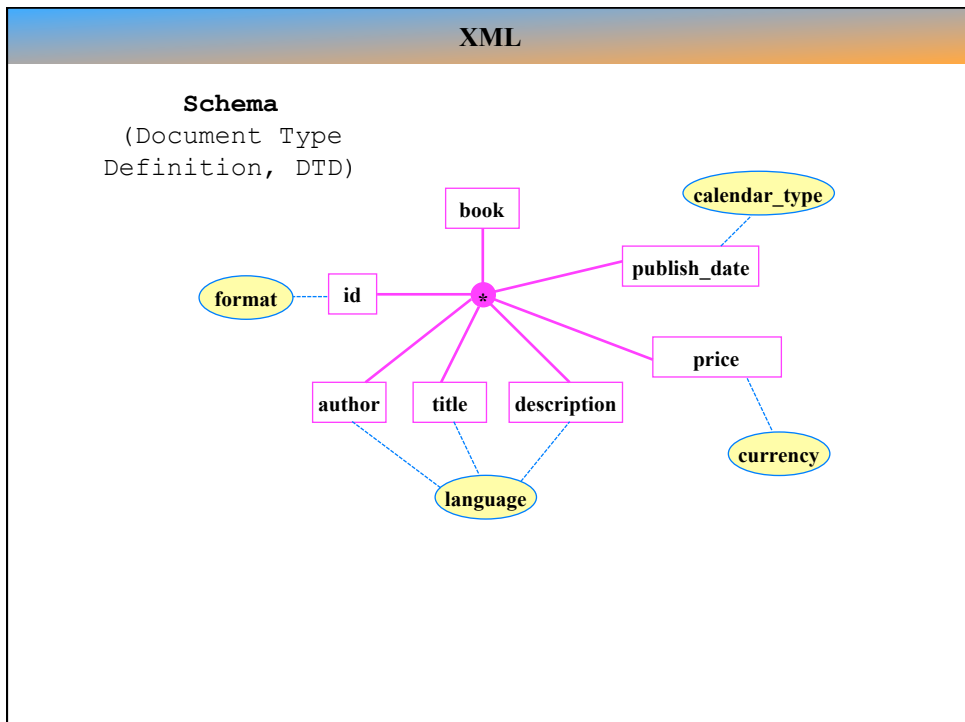


```

<Book Catalogue>
  <book>
    <id>bk101</id>
    <author language="English">Gambardella, Matthew</author>
    <title language="English">XML Developer's Guide</title>
    <price currency="USD">44.95</price>
    <publish_date calendar_type="Gregorian Little Endian">01/10/2000</publish_date>
    <description language="English">An in-depth look at creating applications with XML.</description>
  </book>
  <book>
    <id format="ISBN">817525766-0</id>
    <author language="Greek">Αναγνωστόπουλος, Ιωάννης</author>
    <title language="Greek">Τεχνολογίες Διαδικτύου</title>
    <price currency="Euro">88.00</price>
    <publish_date calendar_type="Gregorian Little Endian">01/03/2009</publish_date>
    <description language="Greek">Βασικές αρχές τεχνολογιών Διαδικτύου και προγραμματισμού.</description>
  </book>
</Book Catalogue>

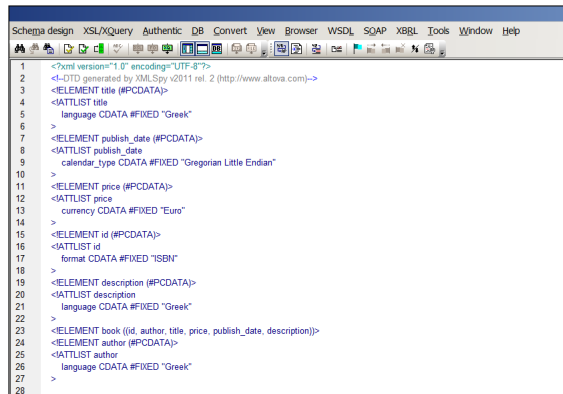
```

**File Format
(Instance)**



XML

DTD instance (powered by Altova XML Spy)



```
Schema design XSL/XQuery Authentic DB Convert View Browser WSDL SOAP XBL Tools Window Help
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!DOCTYPE generated by XML Spy v2011 rel. 2 (http://www.altova.com)->
3 <ELEMENT title (#PCDATA)>
4 <ATTLIST title
5   language CDATA #FIXED "Greek"
6 >
7 <ELEMENT publish_date (#PCDATA)>
8 <ATTLIST publish_date
9   calendar_type CDATA #FIXED "Gregorian Little Endian"
10 >
11 <ELEMENT price (#PCDATA)>
12 <ATTLIST price
13   currency CDATA #FIXED "Euro"
14 >
15 <ELEMENT id (#PCDATA)>
16 <ATTLIST id
17   format CDATA #FIXED "ISBN"
18 >
19 <ELEMENT description (#PCDATA)>
20 <ATTLIST description
21   language CDATA #FIXED "Greek"
22 >
23 <ELEMENT book ((id, author, title, price, publish_date, description))->
24 <ATTLIST book (#PCDATA)>
25 <ATTLIST author
26   language CDATA #FIXED "Greek"
27 >
28 >
```

XML

DTD

- Document Type Definition (DTD):
 - Προτάθηκε σαν μια γραμματική για τα XML έγγραφα
 - Σε κάποιο βαθμό μπορεί να θεωρηθεί ως σχήμα για δεδομένα μορφοποιημένα σε XML

Δηλώσεις DTD

Internal DOCTYPE declaration

<!DOCTYPE root-element [element-declarations]>

External DOCTYPE declaration

<!DOCTYPE root-element SYSTEM "filename">

XML

Internal DOCTYPE declaration

`<!DOCTYPE root-element [element-declarations]>`

```
<?xml version="1.0"?>
<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to (#PCDATA)>
  <!ELEMENT from (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body (#PCDATA)>]
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend</body>
</note>
```

Diagram annotations:

- A blue arrow points from the text "root-element" to the opening tag of the DOCTYPE declaration: `<!DOCTYPE note [`.
- A blue bracket on the right side groups the five element declarations: `<!ELEMENT note (to,from,heading,body)>`, `<!ELEMENT to (#PCDATA)>`, `<!ELEMENT from (#PCDATA)>`, `<!ELEMENT heading (#PCDATA)>`, and `<!ELEMENT body (#PCDATA)>`. This group is labeled "element declarations".

XML

External DOCTYPE declaration

`<!DOCTYPE root-element SYSTEM "filename">`

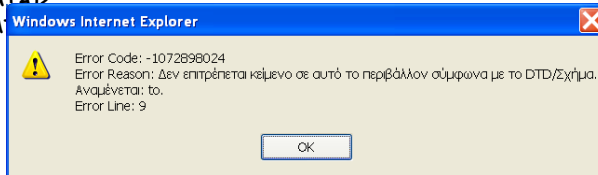
```
<?xml version="1.0"?>
<!DOCTYPE note SYSTEM "note.dtd">
<note>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

note.dtd

```
<!ELEMENT note (to,from,heading,body)>
<!ELEMENT to (#PCDATA)>
<!ELEMENT from (#PCDATA)>
<!ELEMENT heading (#PCDATA)>
<!ELEMENT body (#PCDATA)>
```

Validation errors - DTD

```
<?xml version="1.0" ?>
<!DOCTYPE note [
  <!ELEMENT note (to,from,heading,body)>
  <!ELEMENT to (#PCDATA)>
  <!ELEMENT from (#PCDATA)>
  <!ELEMENT heading (#PCDATA)>
  <!ELEMENT body (#PCDATA)>
]>
<note>
  <date>1/12/2008</date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget our meeting this weekend!</body>
</note>
```



XML

- Elements
 - Μπορούν να περιέχουν κείμενο, άλλα elements ή να είναι άδεια
 - HTML element: <body>some text</body>
 - XML element: <message>some text</message>
 - Empty element: <hr>
- Attributes
 - Παρέχουν επιπλέον πληροφορία σχετικά με τα elements
 - Τοποθετούνται πάντοτε εσωτερικά ενός element
 - Συντάσσονται σε ζεύγη (name/value)
 - Παράδειγμα:
- PCDATA (για Elements)
 - Parsed character data
 - Εξετάζεται από τον parser
 - Δεν πρέπει να περιέχει χαρακτήρες: &, <, ή >. Πρέπει να αντικαθίστανται με &, <, και >
- CDATA (για Attributes)
 - Character data

XML

Δήλωση elements

- `<!ELEMENT element-name category>`
ή
`<!ELEMENT element-name (element-content)>`
- Empty element
`<!ELEMENT element-name EMPTY>`
- Element με PCDATA
`<!ELEMENT element-name (#PCDATA)>`
Παράδειγμα: `<!ELEMENT from (#PCDATA)>`
- Element με οποιοδήποτε περιεχόμενο
`<!ELEMENT element-name ANY>`
`<!ELEMENT note ANY>`
- Elements με elements-παιδιά
`<!ELEMENT element-name (child1)>`
ή
`<!ELEMENT element-name (child1,child2,...)>`
Παράδειγμα: `<!ELEMENT note (to,from,heading,body)>`
Τα elements-παιδιά πρέπει να δηλώνονται με την ακριβή σειρά εμφάνισής τους

XML

Δήλωση elements

- Ακριβώς μία εμφάνιση element
`<!ELEMENT element-name (child-name)>`
Παράδειγμα: `<!ELEMENT note (message)>`
Το element message πρέπει να υπάρχει ακριβώς μία φορά εντός του element note
- Τουλάχιστον μία εμφάνιση
`<!ELEMENT element-name (child-name+)>`
Παράδειγμα: `<!ELEMENT note (message+)>`
- Προαιρετική εμφάνιση-όσες φορές χρειαστεί
`<!ELEMENT element-name (child-name*)>`
Παράδειγμα: `<!ELEMENT student (course_left*)>`
- Προαιρετική εμφάνιση-μονάχα μία φορά
`<!ELEMENT element-name (child-name?)>`
`<!ELEMENT student (diploma_thesis?)>`
- Είτε/είτε περιεχόμενο
Παράδειγμα: `<!ELEMENT note (to,from,header,(message|body))>`

XML

Δήλωση attributes

- `<!ATTLIST element-name attribute-name attribute-type default-value>`
 Παράδειγμα:
 XML κώδικας: `<payment type="check" />`
 DTD: `<!ATTLIST payment type CDATA "check">`
- Attribute type (παράδειγματα)
 CDATA → Character data
 (en1|en2|..) → Τιμή από μία αριθμημένη λίστα
 ID → Μοναδικό id
- Attribute value
 Value → Η προκαθορισμένη τιμή του attribute
 #REQUIRED To attribute περιλαμβάνεται υποχρεωτικά
 #IMPLIED → Το attribute δεν περιλαμβάνεται υποχρεωτικά
 #FIXED value → Η τιμή του attribute είναι προκαθορισμένη και σταθερή

DTD:

```
<!ATTLIST contact fax CDATA #IMPLIED>
Εγκυρο XML:
1. <contact fax="555-667788" />
2. <contact />
```

DTD:

```
<!ATTLIST person number
CDATA #REQUIRED>
Εγκυρο XML:
<person number="5677" />
Άκυρο XML:
<person />
```

DTD:

```
<!ATTLIST sender company CDATA #FIXED "Microsoft">
Εγκυρο XML:
<sender company="Microsoft" />
Άκυρο XML:
<sender company="W3Schools" />
```

Άλλος ορισμός default values σε attribute

```
<!ATTLIST element-name attribute-name (en1|en2|..) default-value>
<!ATTLIST payment type (check|cash) "cash">
```

XML

Δήλωση attributes στο DTD trompette.dtd

- `<!ELEMENT product ((name, price, address))>`
- `<!ELEMENT name (#PCDATA)>`
- `<!ATTLIST name`
 `language CDATA #FIXED "French">`
- `<!ELEMENT price (#PCDATA)>`
- `<!ATTLIST price`
 `currency CDATA #FIXED "Euro">`
- `<!ELEMENT address ((street, zip))>`
- `<!ATTLIST address`
 `format CDATA #FIXED "XL1245"`
 `language CDATA #FIXED "French">`
- `<!ELEMENT street (#PCDATA)>`
- `<!ELEMENT zip (#PCDATA)>`

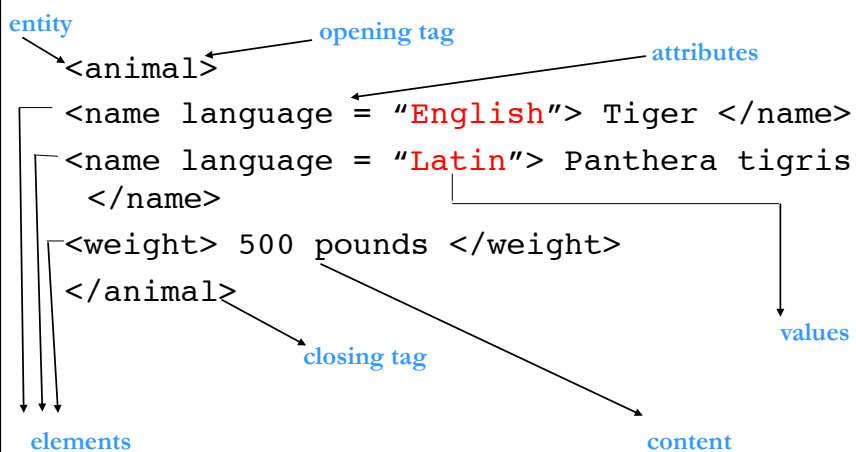
XML

Ο κώδικας

Trompette.xml

- `<product>`
- `<name language="French">trompette no 6</name>`
- `<price currency="Euro"> 420.12 </price>`
- `<address format="XL1245" language="French">`
- `<street>31 rue Croix-Bosset</street>`
- `<zip>92874</zip>`
- `</address>`
- `</product>`

XML



```
<picture filename = "tiger.jpg" />
```

empty element, just an attribute

Tiger.xml

32

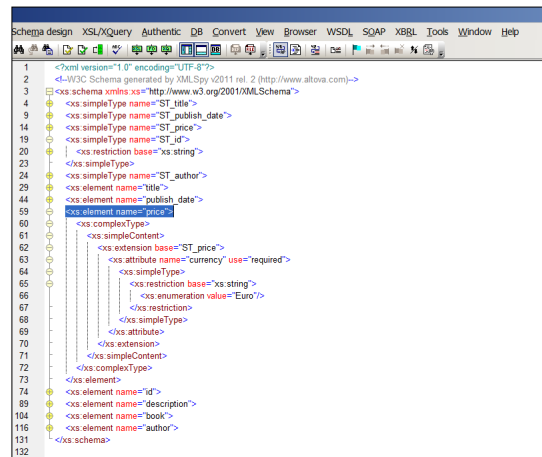
XML

```
<!ELEMENT animal ((name+, weight))>
<!ELEMENT weight (#PCDATA)>
<!ELEMENT name (#PCDATA)>
<!ATTLIST name
  language (Latin | English) #REQUIRED>
```

Tiger.dtd

XML

XSD instance (powered by Altova XML Spy)



```
1 <?xml version="1.0" encoding="UTF-8"?>
2 <!-- Schema generated by XMLSpy v2011 rel. 2 (http://www.altova.com) -->
3 <xs:schema xmlns:xs="http://www.w3.org/2001/XMLSchema">
4   <xs:simpleType name="ST_title">
5     <xs:base base="xs:string"/>
6   </xs:simpleType>
7   <xs:simpleType name="ST_publish_date">
8     <xs:base base="xs:string"/>
9   </xs:simpleType>
10  <xs:simpleType name="ST_price">
11    <xs:base base="xs:string"/>
12  </xs:simpleType>
13  <xs:simpleType name="ST_id">
14    <xs:base base="xs:string"/>
15  </xs:simpleType>
16  <xs:simpleType name="ST_author">
17    <xs:base base="xs:string"/>
18  </xs:simpleType>
19  <xs:element name="title">
20    <xs:simpleContent/>
21    <xs:base base="ST_title"/>
22  </xs:element>
23  <xs:element name="publish_date">
24    <xs:simpleContent/>
25    <xs:base base="ST_publish_date"/>
26  </xs:element>
27  <xs:element name="price">
28    <xs:simpleContent/>
29    <xs:base base="ST_price"/>
30    <xs:attribute name="currency" use="required"/>
31  </xs:element>
32  <xs:element name="id">
33    <xs:simpleContent/>
34    <xs:base base="ST_id"/>
35  </xs:element>
36  <xs:element name="description">
37    <xs:simpleContent/>
38    <xs:base base="xs:string"/>
39  </xs:element>
40  <xs:element name="book">
41    <xs:complexType/>
42    <xs:base base="xs:string"/>
43    <xs:restriction base="xs:string">
44      <xs:enumeration value="Euro"/>
45    </xs:restriction>
46  </xs:element>
47  <xs:element name="author">
48    <xs:simpleContent/>
49    <xs:base base="ST_author"/>
50  </xs:element>
51 </xs:schema>
```