XMLTree Model



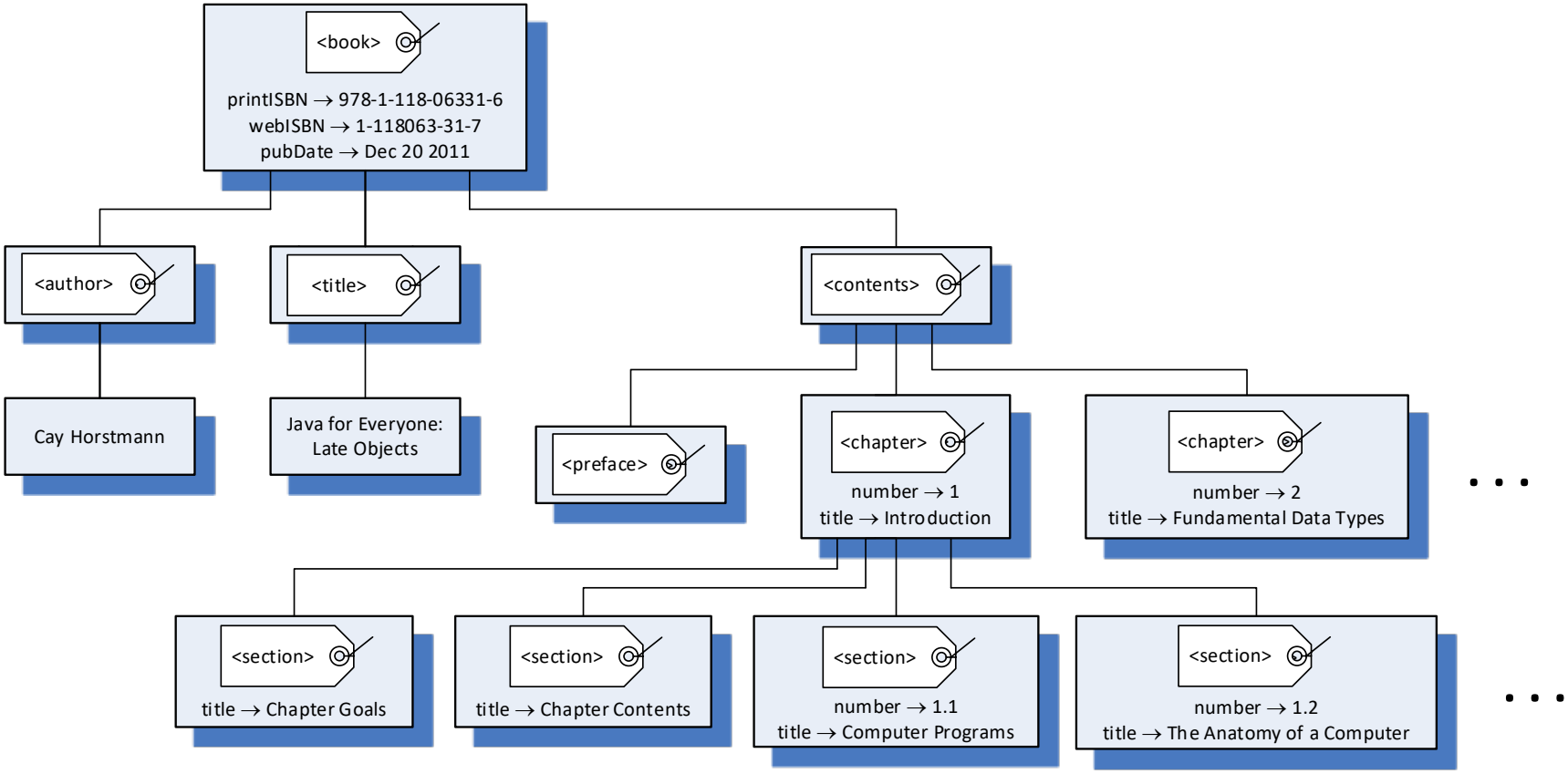

`XMLTree`

- The ***`XMLTree`*** component family allows you to create, and navigate through, a tree whose structure mirrors that of an XML file
  - The file from which the tree is created may come from your computer or from the web
  - You need not worry about ***parsing*** the file (recognizing tags, matching start-end tags, identifying attributes, etc.), as this is done for you by `XMLTree`

Example XML File

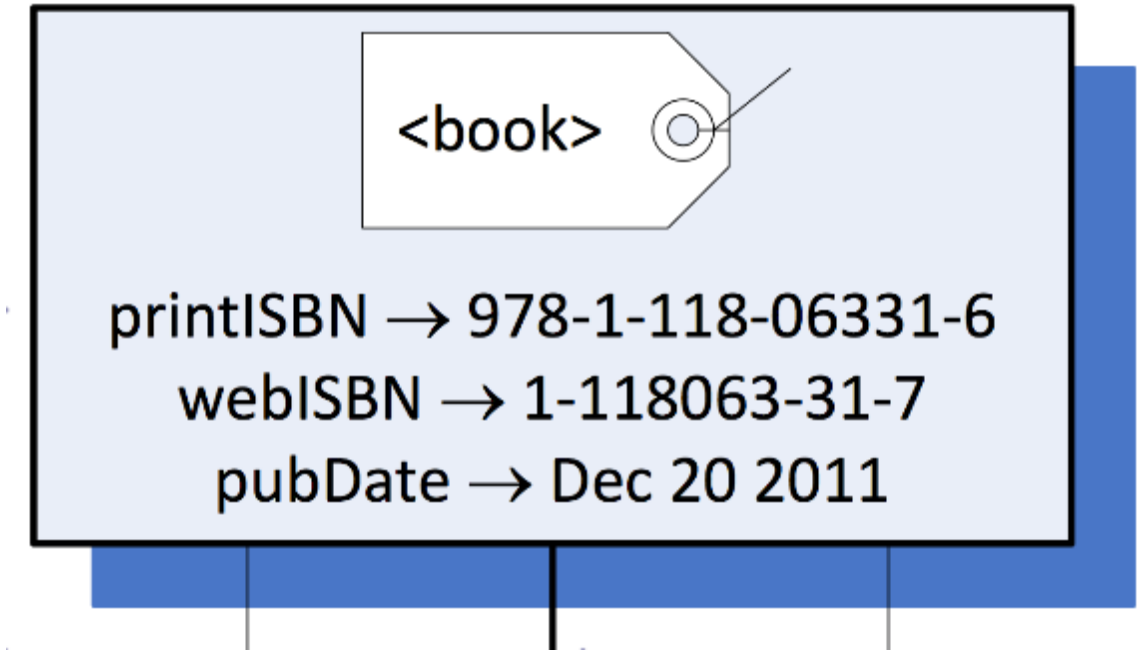
```
<?xml version="1.0" encoding="UTF-8"?>
<book printISBN="978-1-118-06331-6" webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  <author>Cay Horstmann</author>
  <title>Java for Everyone: Late Objects</title>
  <contents>
    <preface></preface>
    <chapter number="1" title="Introduction">
      <section title="Chapter Goals"></section>
      <section title="Chapter Contents"></section>
      <section title="Computer Programs" number="1.1"></section>
      <section title="The Anatomy of a Computer" number="1.2"></section>
      ...
    </chapter>
    <chapter number="2" title="Fundamental Data Types">
    </chapter>
    ...
  </contents>
</book>
```

Example `XMLTree` Created



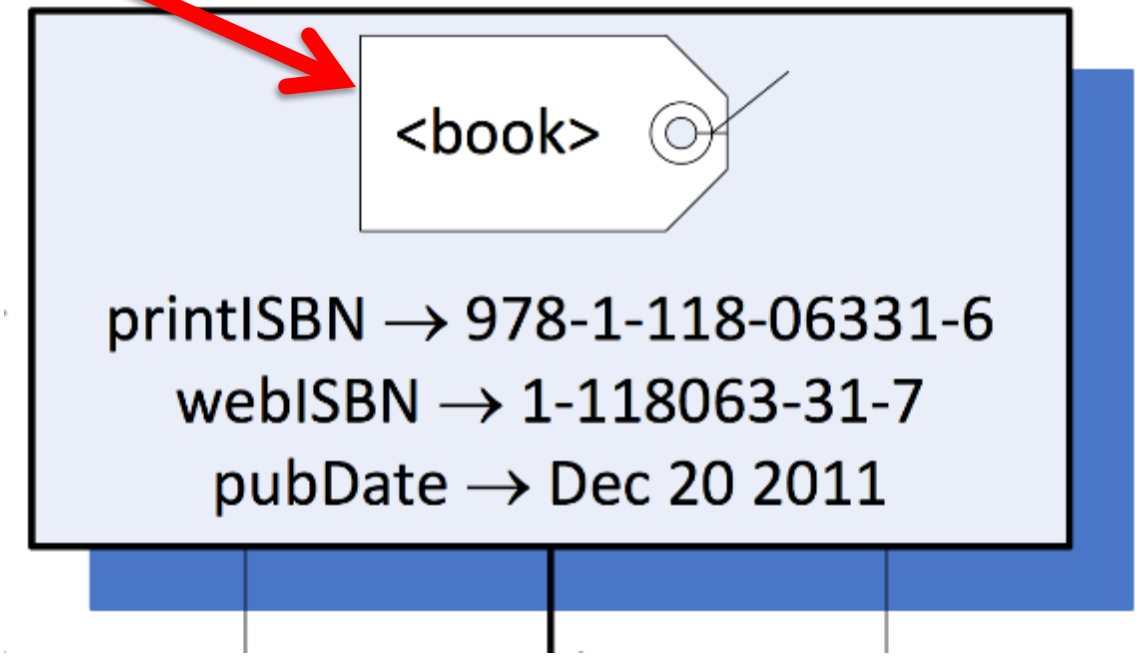
Top-Level Element = Root of Tree

```
<book printISBN="978-1-118-06331-6"
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  ...
</book>
```



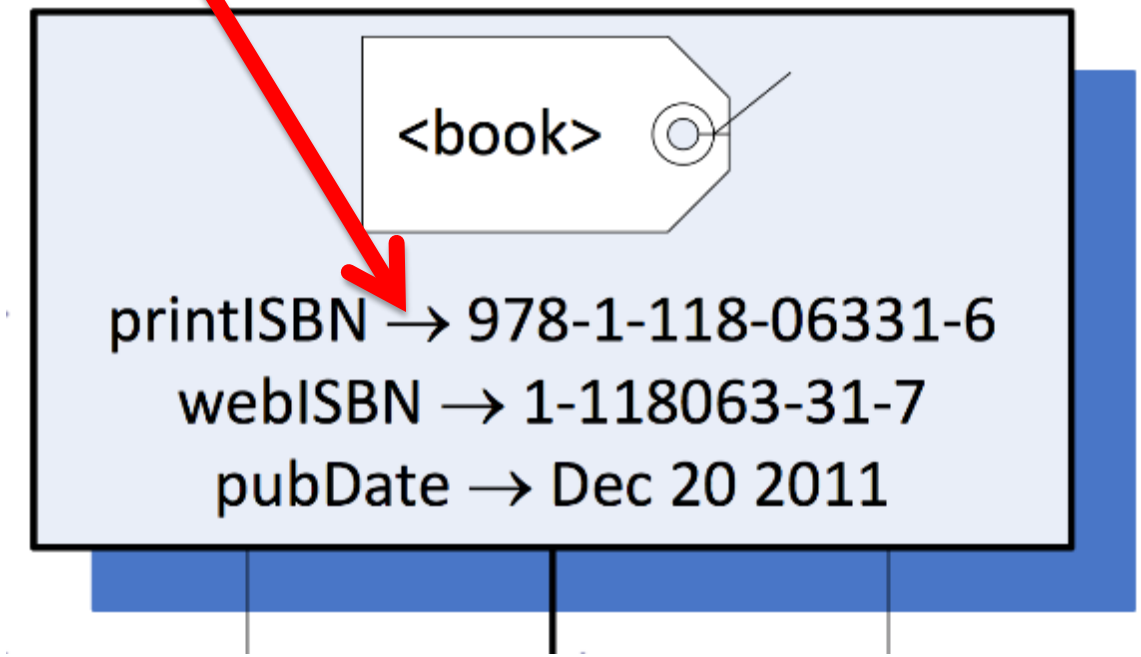
Top-Level Element = Root of Tree

```
<book printISBN="978-1-118-06331-6"
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  ...
</book>
```



Top-Level Element = Root of Tree

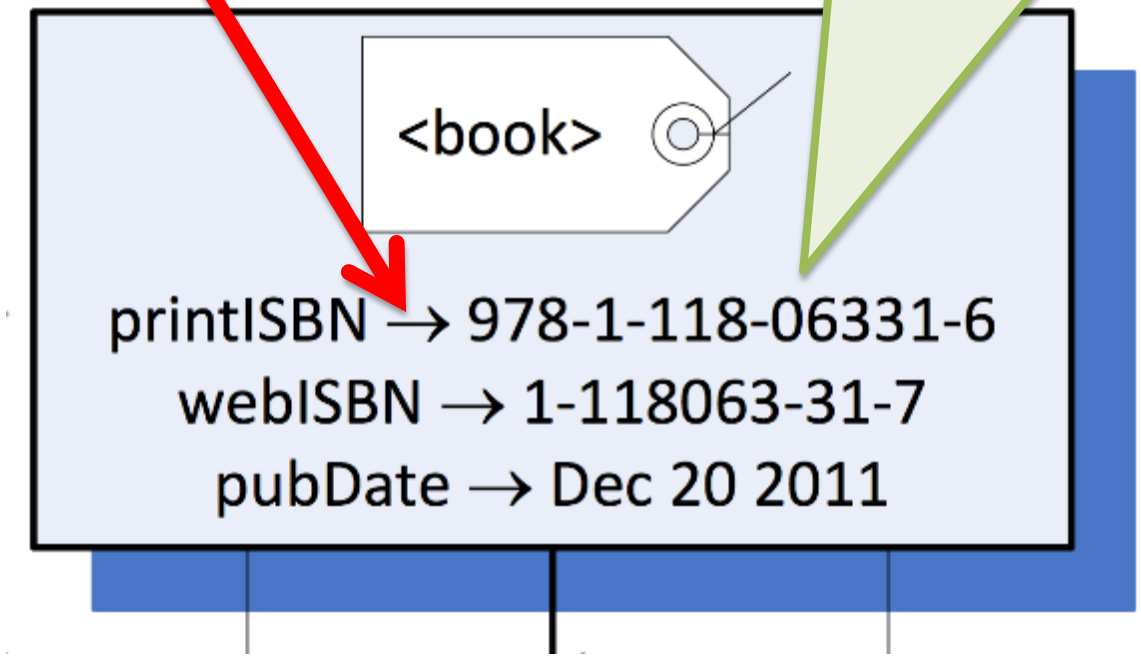
```
<book printISBN="978-1-118-06331-6"
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  ...
</book>
```



Top-Level Element = Root of Tree

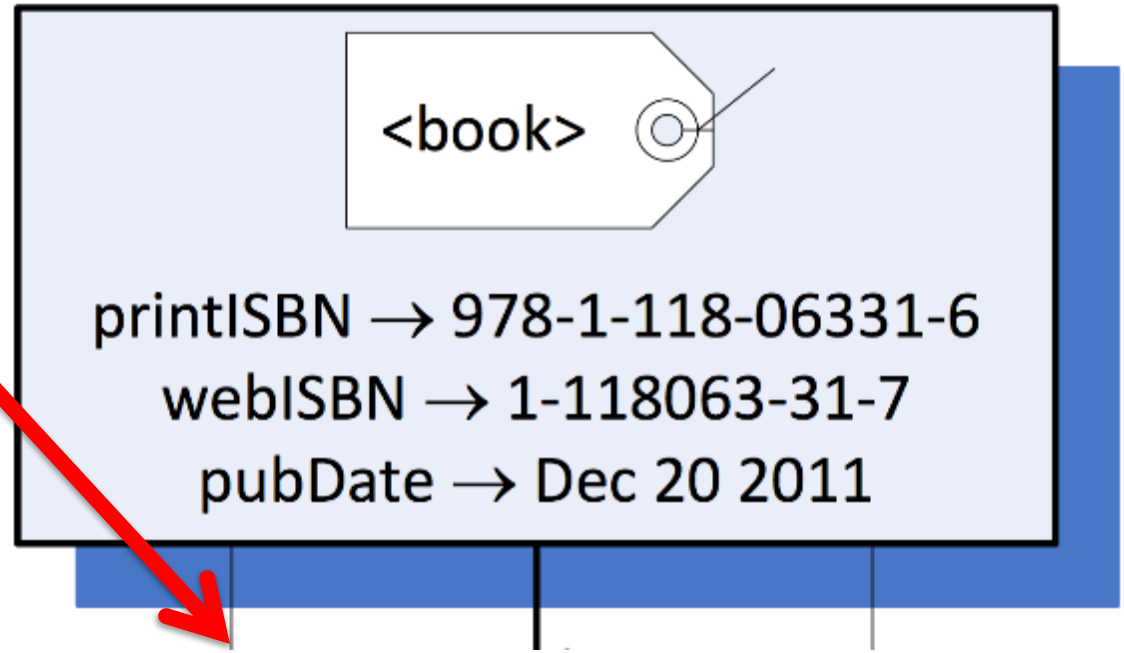
```
<book printISBN="978-1-
  webISBN="1-118063-31-
  pubDate="Dec 20 2011"
  ...
</book>
```

The attribute name-value pairs of an element/tag are *in the tree node* of that element/tag.



Top-Level Element = Root of Tree

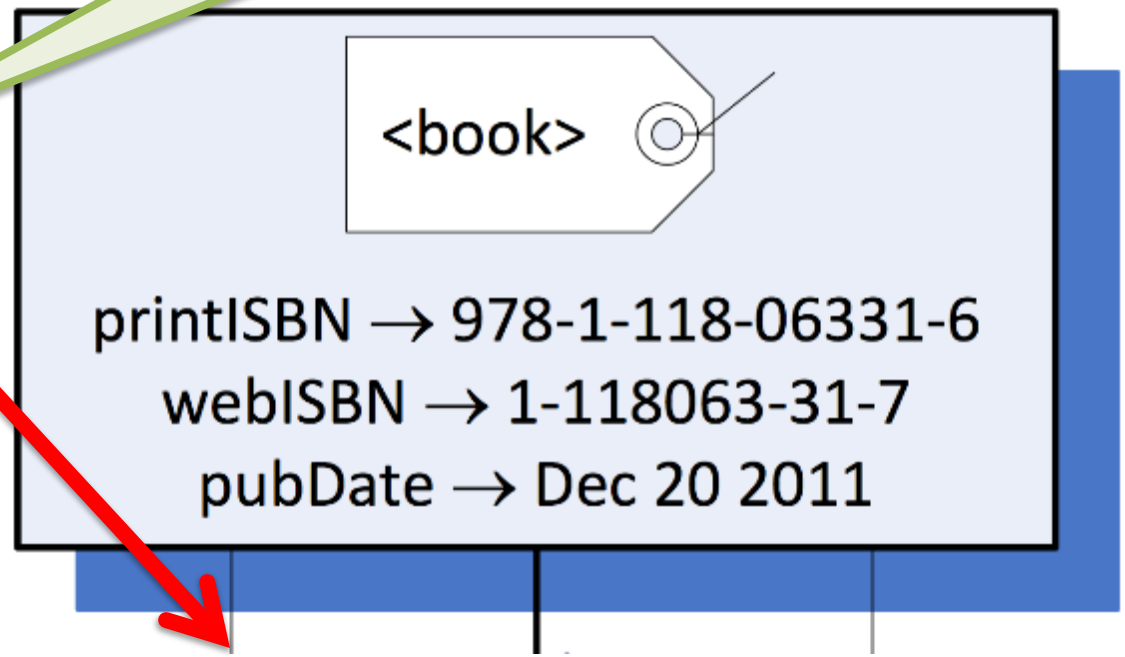
```
<book printISBN="978-1-118-06331-6"
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  . . .
</book>
```



Top-Level Element = Root of Tree

```
<book printISBN="978-1-118
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011"
  ...
</book>
```

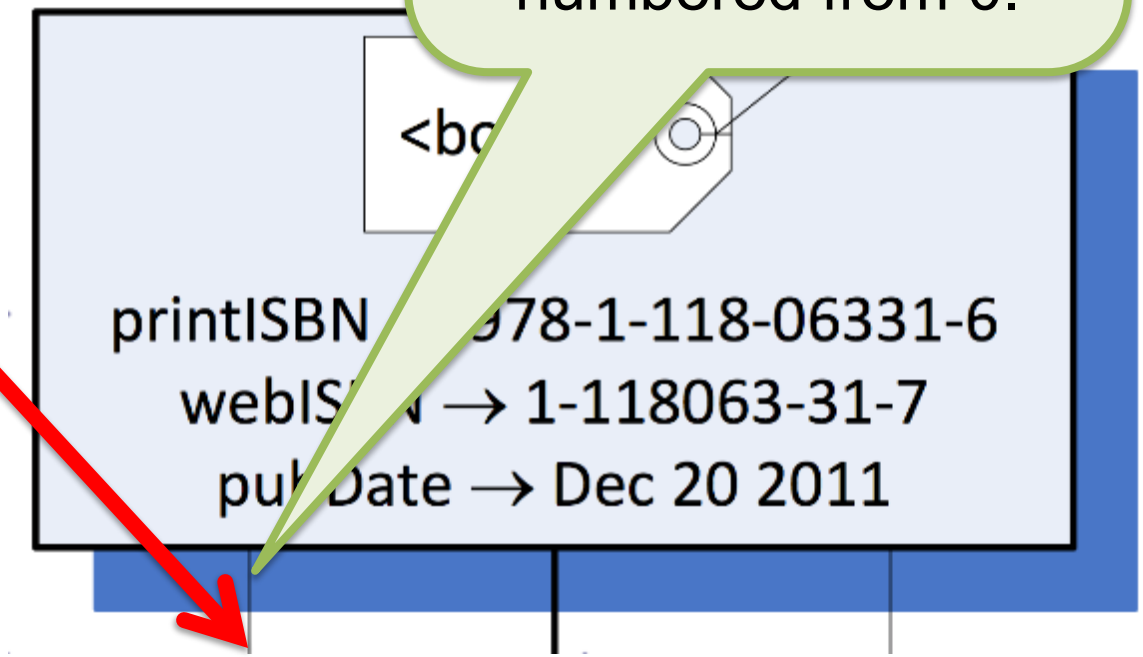
The children of the top-level element/tag are the children of the root of the tree.



Top-Level Element = Root of Tree

```
<book printISBN="978-1-11
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  . . .
</book>
```

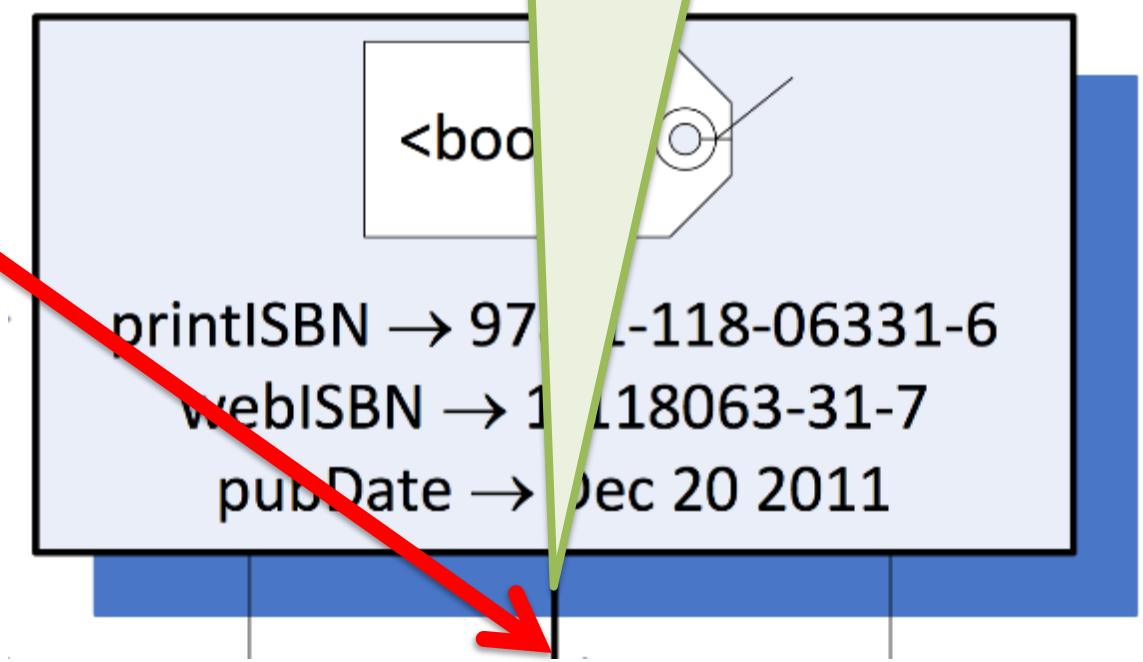
The first child node of the root is known as "child 0" because children are numbered from 0.



Top-Level Element = Root of Tree

```
<book printISBN="978-1-11
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  . . .
</book>
```

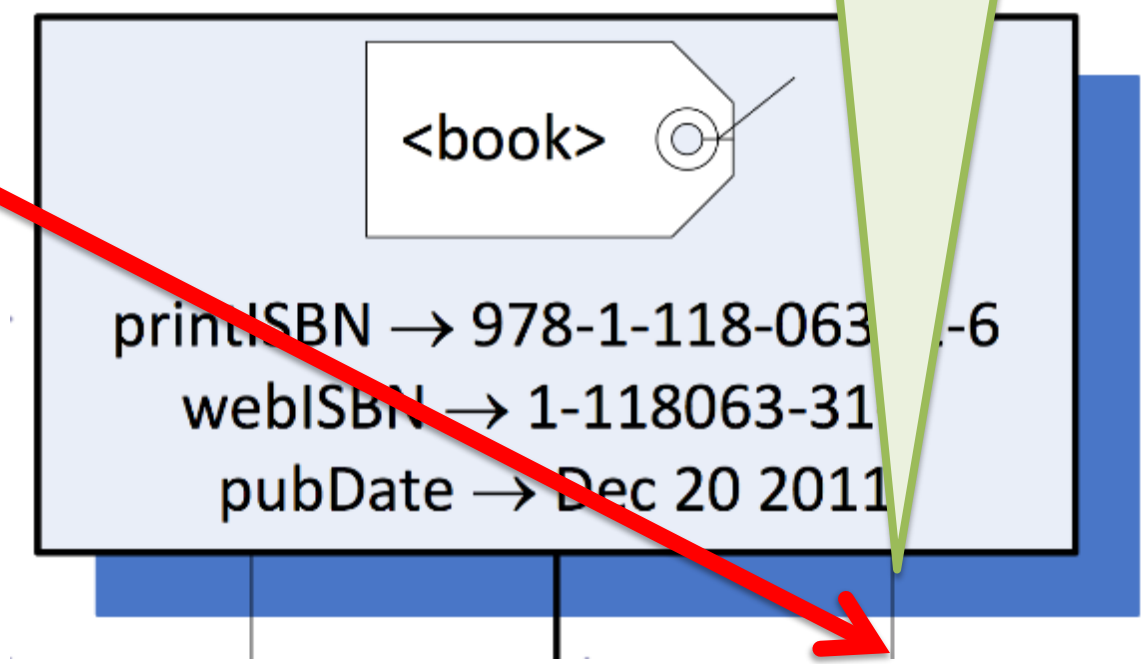
The second child node of the root is known as "child 1".



Top-Level Element = Root of Tree

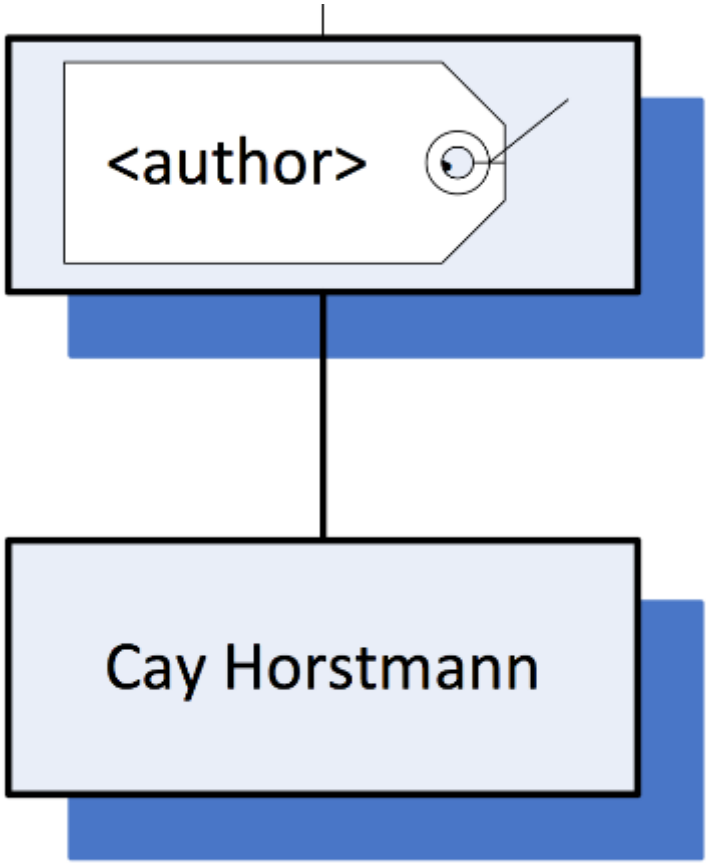
```
<book printISBN="978-1-11
  webISBN="1-118063-31-7"
  pubDate="Dec 20 2011">
  . . .
</book>
```

The third child node of the root is known as "child 2".



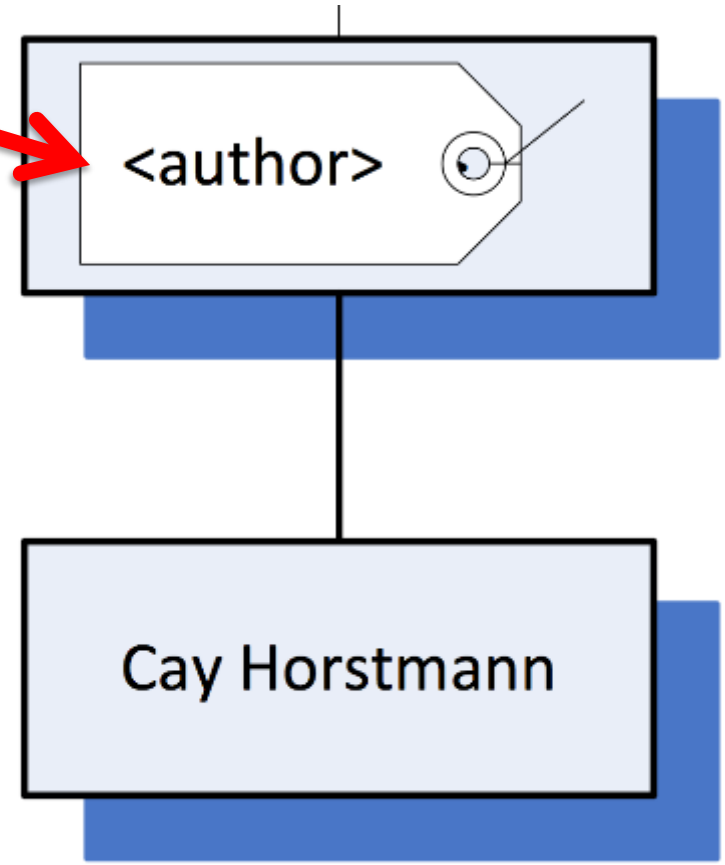
First Child Element = First Child Node

```
<book ...>
    <author>Cay Horstmann</author>
    ...
</book>
```



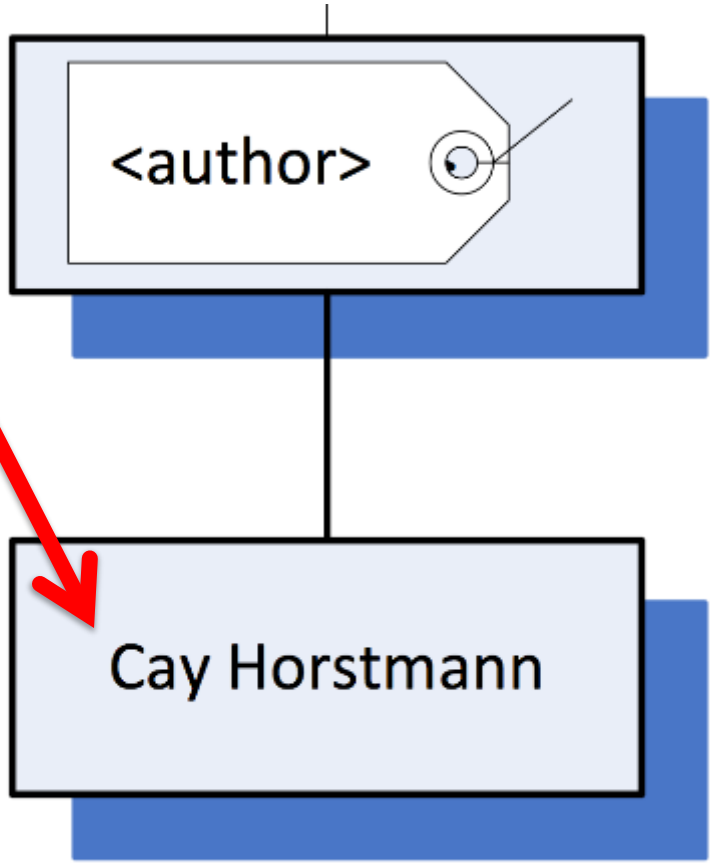
First Child Element = First Child Node

```
<book ...>
    <author>Cay Horstmann</author>
    ...
</book>
```



First Child Element = First Child Node

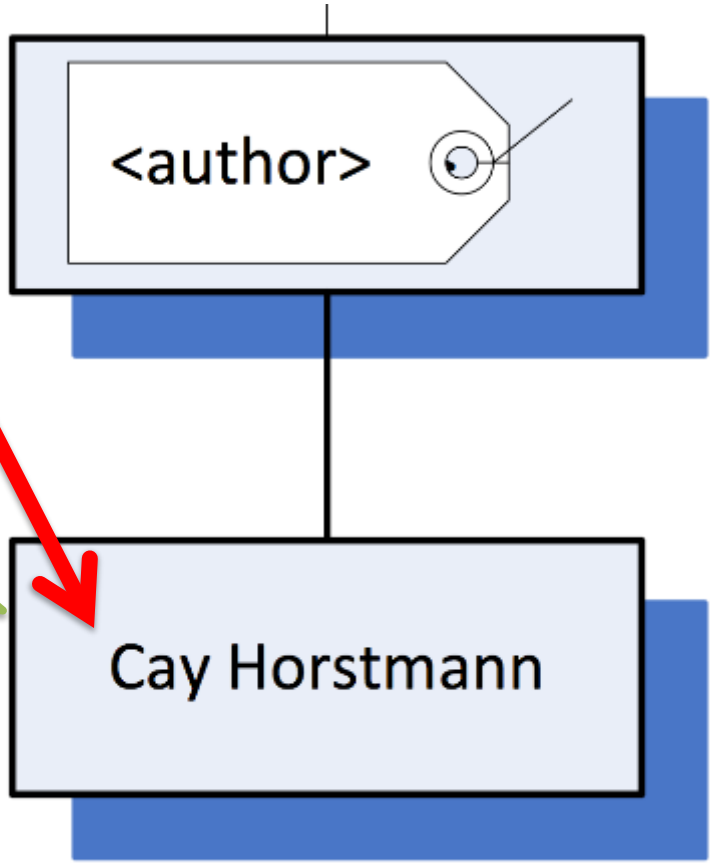
```
<book ...>
    <author>Cay Horstmann</author>
    ...
</book>
```



First Child Element = First Child Node

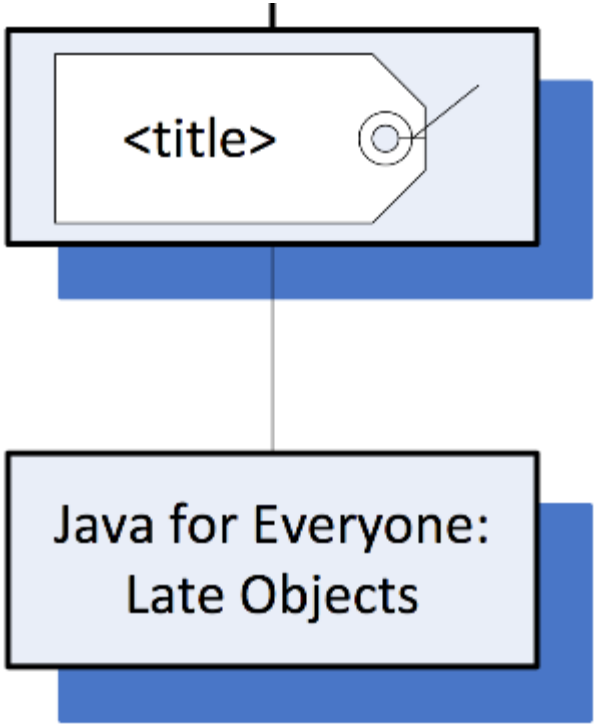
```
<book ...>
    <author>Cay Horstmann</author>
    ...
</book>
```

The content of an element/tag is *in a child node* of the node for that element/tag; the content node itself *has no tag*.



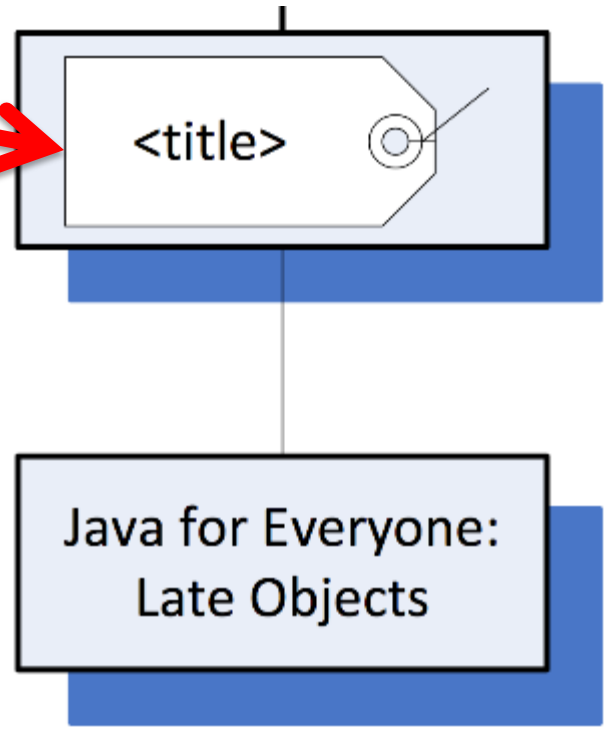
Second Child Element = Second Child Node

```
<book ...>
    ...
    <title>Java for Everyone: Late Objects</title>
    ...
</book>
```



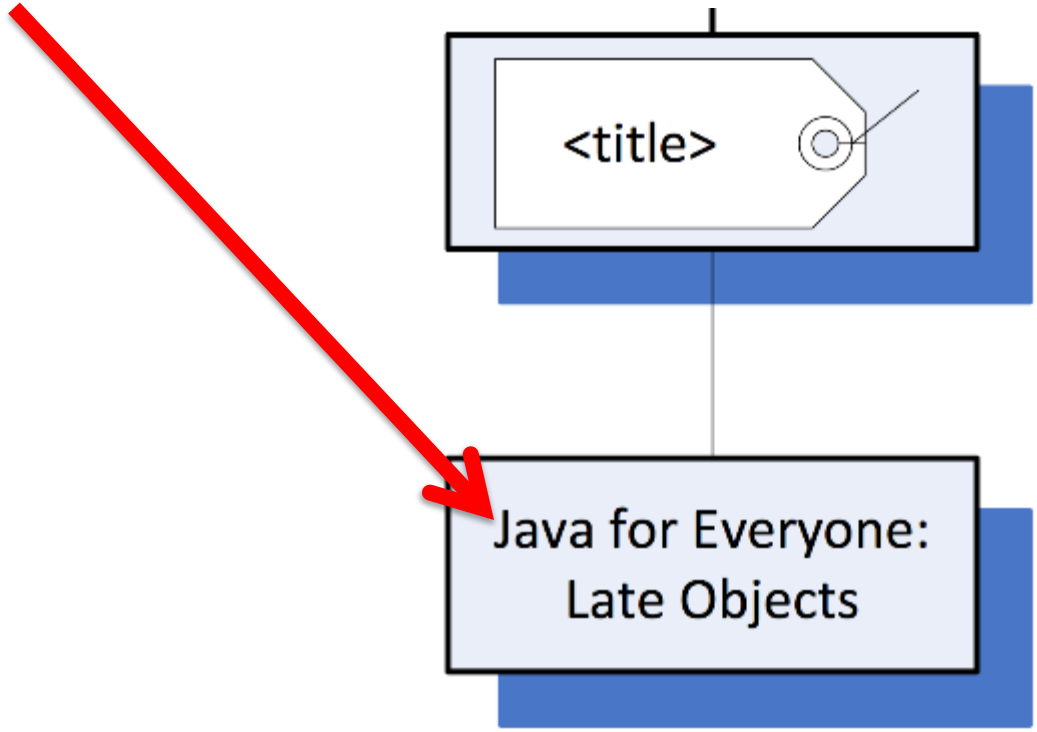
Second Child Element = Second Child Node

```
<book ...>
    ...
    <title>Java for Everyone: Late Objects</title>
    ...
</book>
```



Second Child Element = Second Child Node

```
<book ...>
    ...
    <title>Java for Everyone: Late Objects</title>
    ...
</book>
```



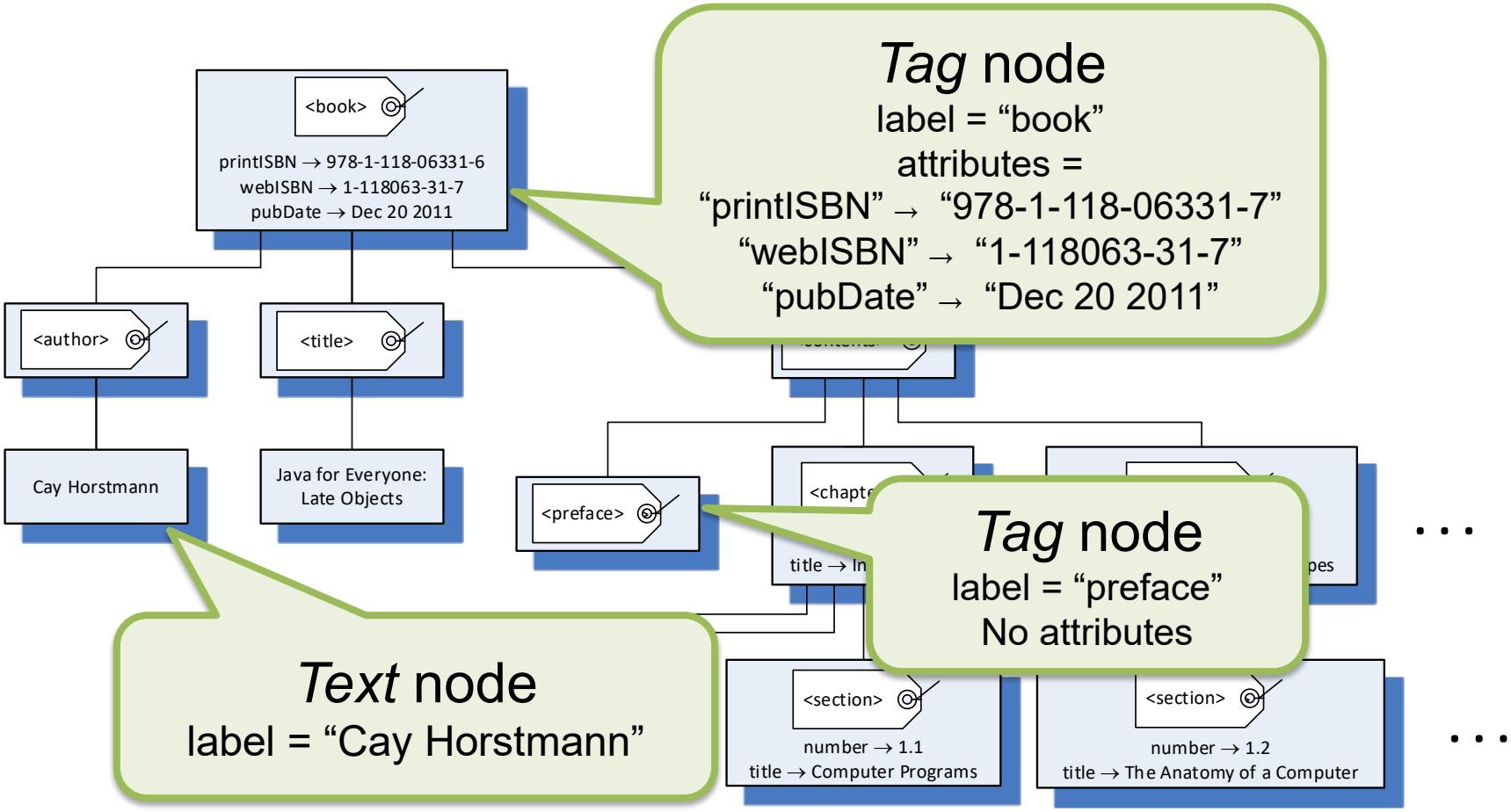
… And So On!

- The same rules apply at all levels of the XML file and of the `XMLTree` object that is created from it

Key Properties/Invariants

- Every node in the tree represents either a *tag* (e.g., <`author`>) or the *text* content (e.g., "`Cay Horstman`") of an element
- Every node in the tree has a ***label*** `String` that is *either* the tag name (e.g., "`author`") *or* the text content itself (e.g., "`Cay Horstman`")
- Only tag nodes can have zero or more attribute name-value pairs, each of which is a `String` (e.g., "`pubDate`" → "`Dec 20 2011`")

Tag Nodes and Text Nodes



Reality Check

- Carefully examine the example XML file and the example `XMLTree` diagram shown earlier to check your understanding of how the file content is ***modeled*** by the tree

Can You Draw The Tree?

```
<?xml version="1.0"?>
<buckeyes>
  <location stadium="home" />
  Go Bucks!
  <game opp="UAB" date="22 Sep 2012">
    <forecast>Sunny</forecast>
  </game>
</buckeyes>
```

Resources

- OSU CSE Components API: `XMLTree`
  - http://web.cse.ohio-state.edu/software/common/doc/