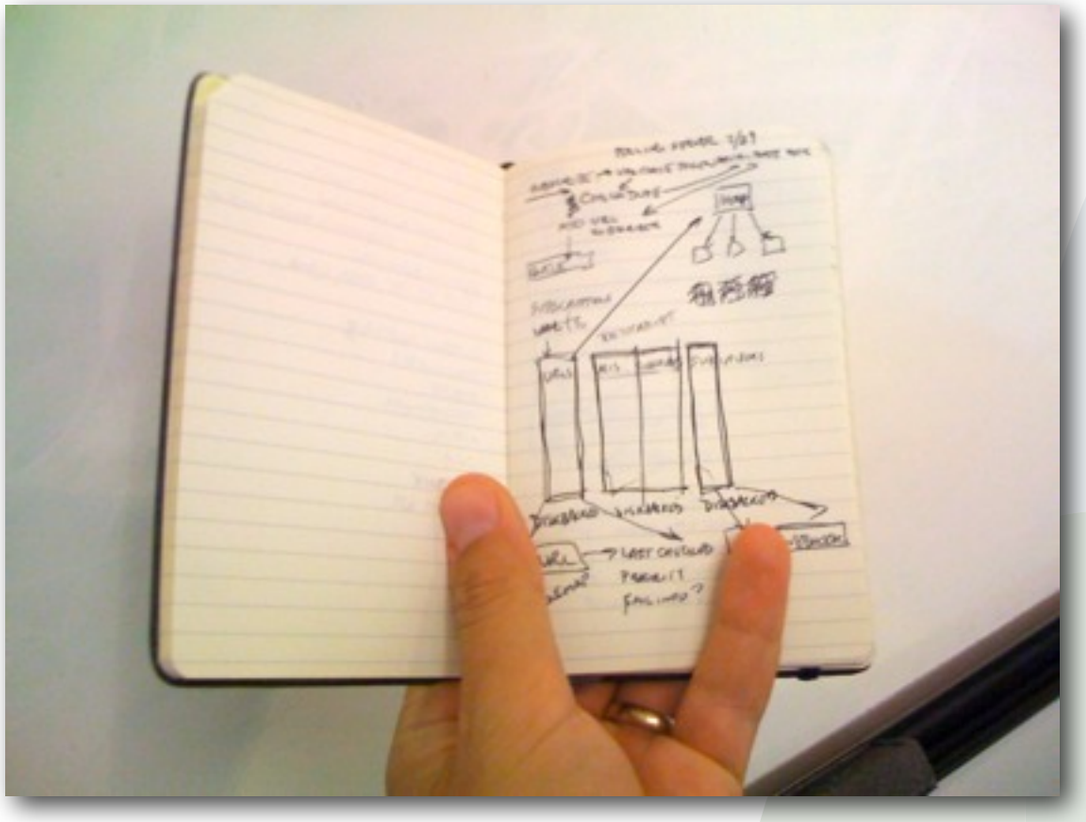
User Interaction: XML and JSON

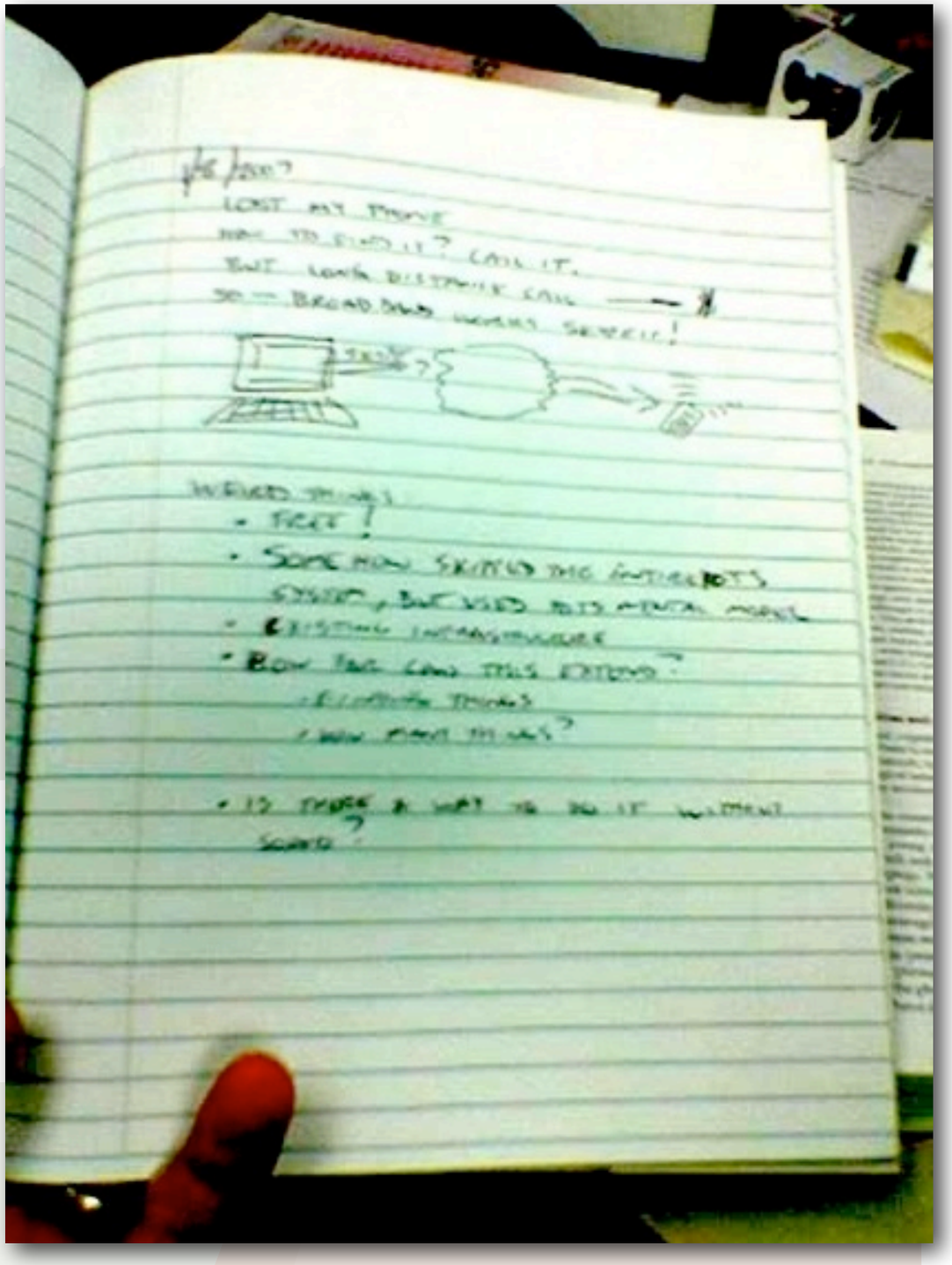
Asst. Professor Donald J. Patterson
INF 133 Fall 2011

What might a design notebook be like?

Cooler →

What does a design notebook entry look like?

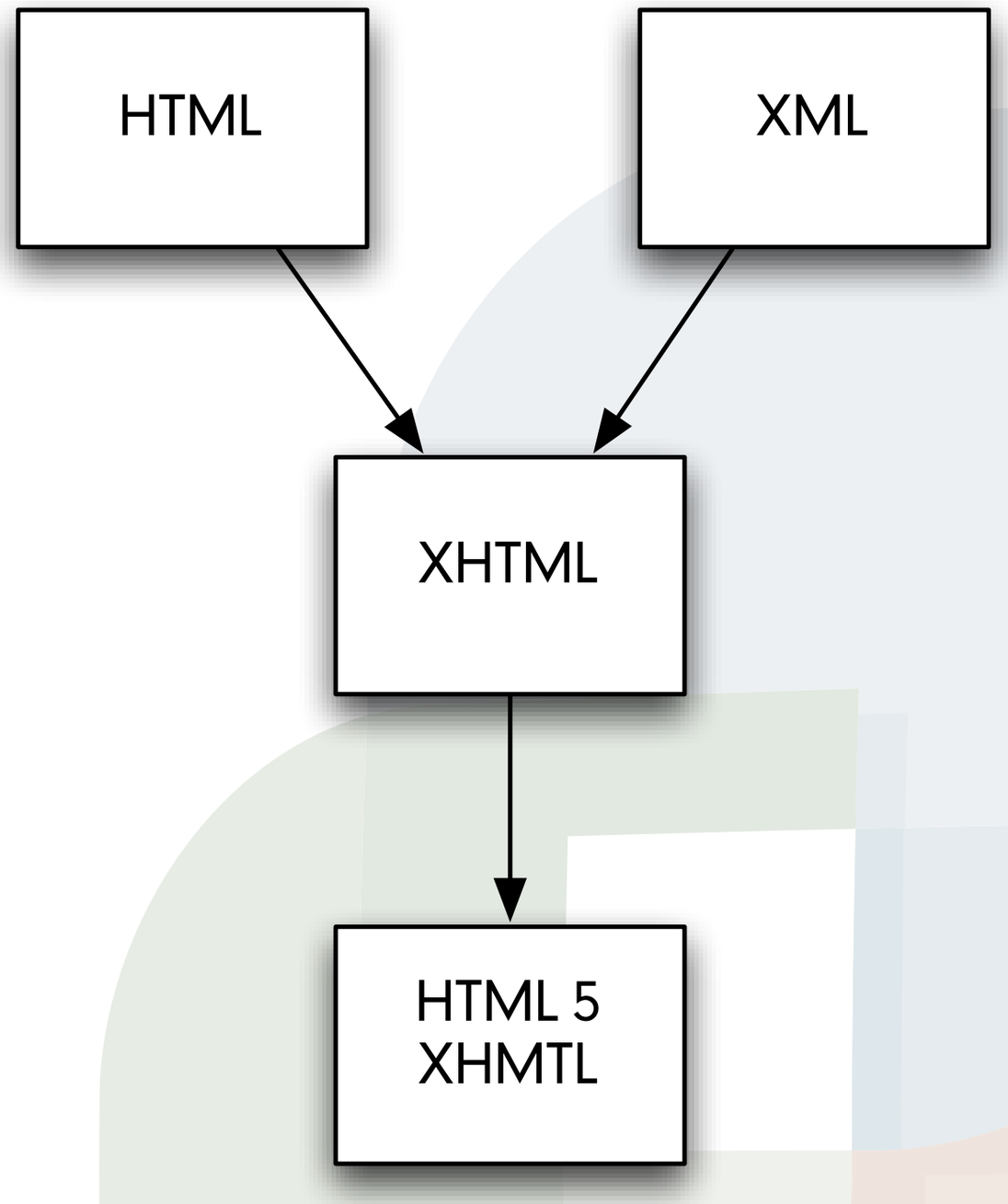




HTML and XML

- 1989: Tim Berners-Lee invents the Web with HTML as its publishing language
- Based on SGML
  - Separates data from presentation
  - No hypertext
- 1993: Mosaic browser is released
- 1994: World Wide Web Consortium is formed
- 1995: HTML 2.0 published IETF
- 1997: HTML 3.2 published by W3C
- 1995: Internet Explorer is released
- 1999: HTML 4.01 standardized and released
- 2000: XHTML standard released
- 2010: HMTL 5 Draft Standard Released

HTML and XML

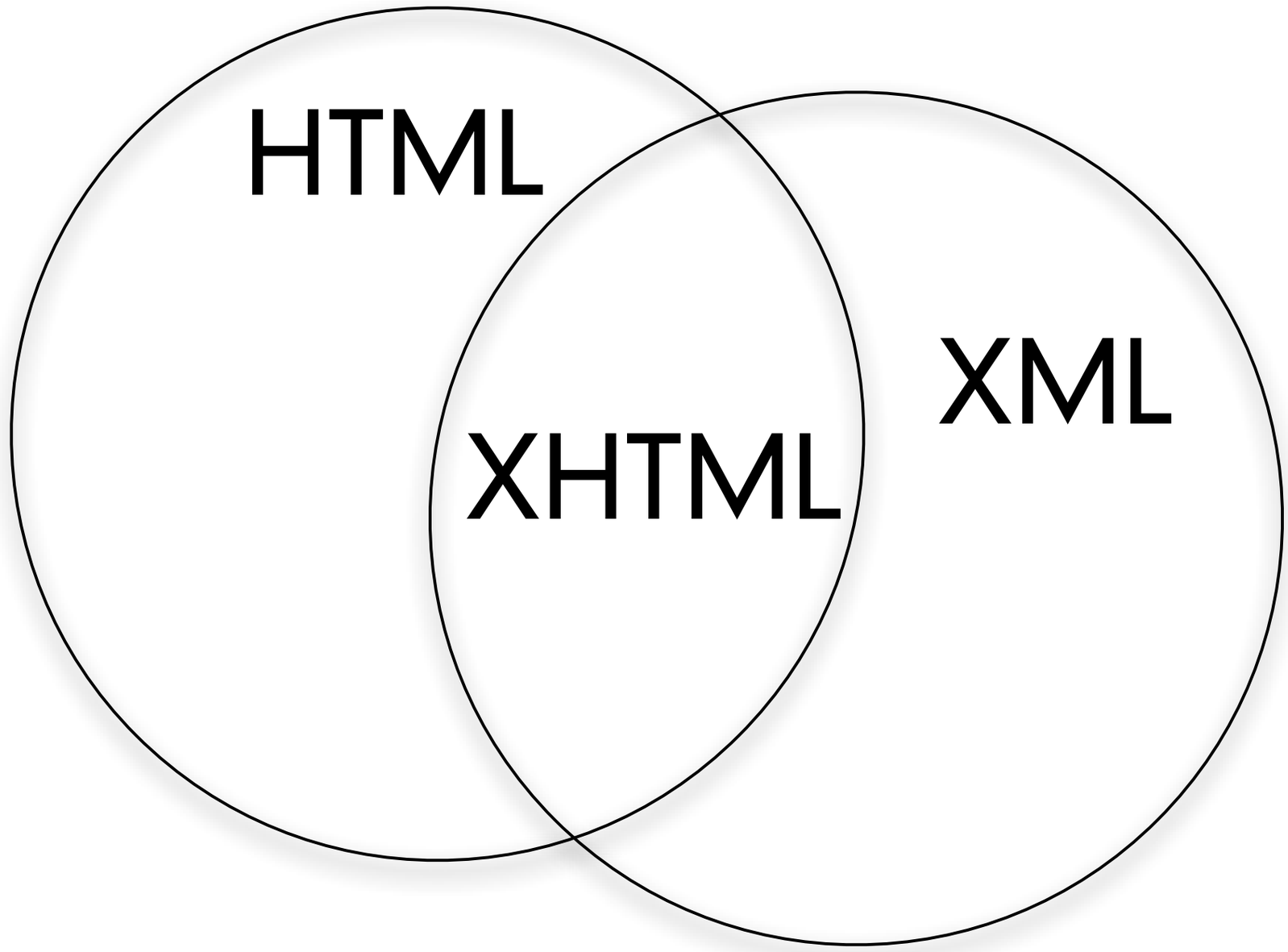


HTML5

- Support for SVG and MathML
- New tags
  - add semantic meaning
    - section
    - article
  - add multimedia processing
    - canvas
    - video
    - audio
- Some tags deleted

- offline storage
- drag and drop
- document editing

HTML and XML



- HTML, XML and JSON
  - Structured Data Formats that evolved with the web
  - Text with a syntax applied
  - They can represent a huge variety of information
  - They enable data transport
    - Different systems and technologies and programming languages depend on the syntax being standardized

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

- What is XML?
  - XML stands for “eXtensible Markup Language”
  - XML was designed to in the context of separating
    - data from display
- XML tags are not predefined
  - You define your own tags
- XML is designed to be self-descriptive

- The Difference Between XML and HTML
- XML
  - designed to transport and store data
  - It looks like HTML
  - The focus is on what the data is
- HTML
  - designed to display data
  - it typically is “broken-XML”
  - XHTML is
    - HTML that conforms to XML standard
  - Traditionally the focus was on how data looks

- XML Does not DO Anything
  - It is a data format
  - A program must be written to manipulate the data
    - To search the data
    - To display the data
    - To change the data
  - Even though the data seems to be associated with a task it is still just data.

Schema

Tags

Characters

- XML is Just Plain Text
  - There is nothing fancy about the storage
  - A program that can read and write text can read and write XML
- an XML-aware application
  - Expects a valid tag structure
  - Interprets the tags in a particular way

Schema

Tags

Characters

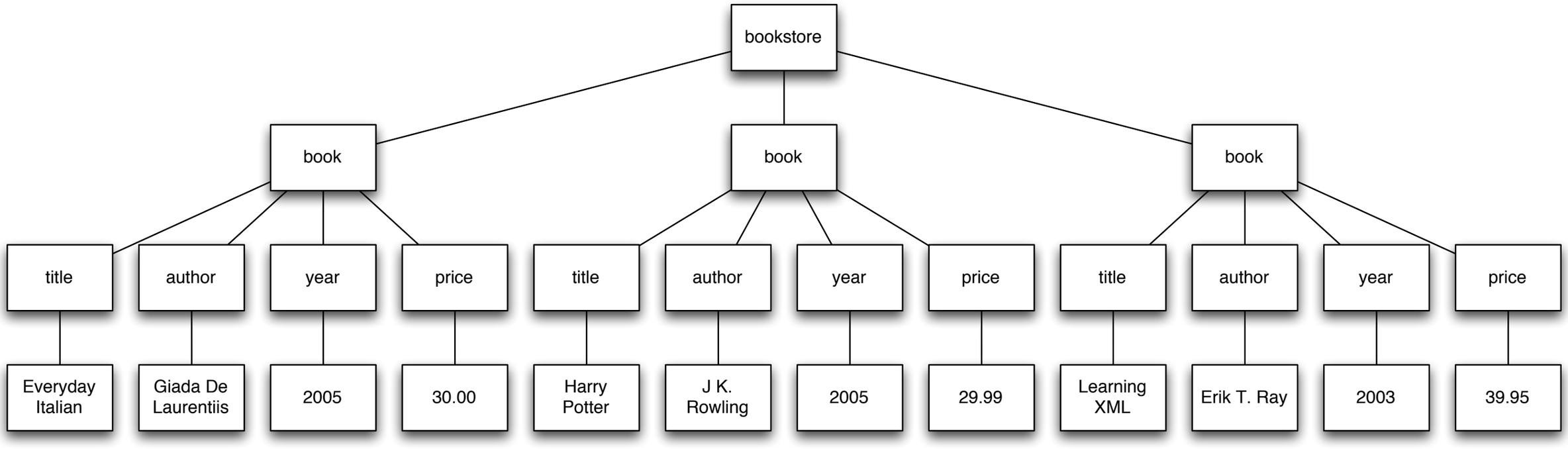
- With XML You Invent Your Own Tags
  - <from> and <to>
    - are not defined anywhere official
    - they are invented by the author
  - There are no predefined tags
- In contrast, HTML has predefined tags
  - <p> <href> etc.,
- In XML the author defines the tags and the structure
  - within the bounds of a “valid XML document”

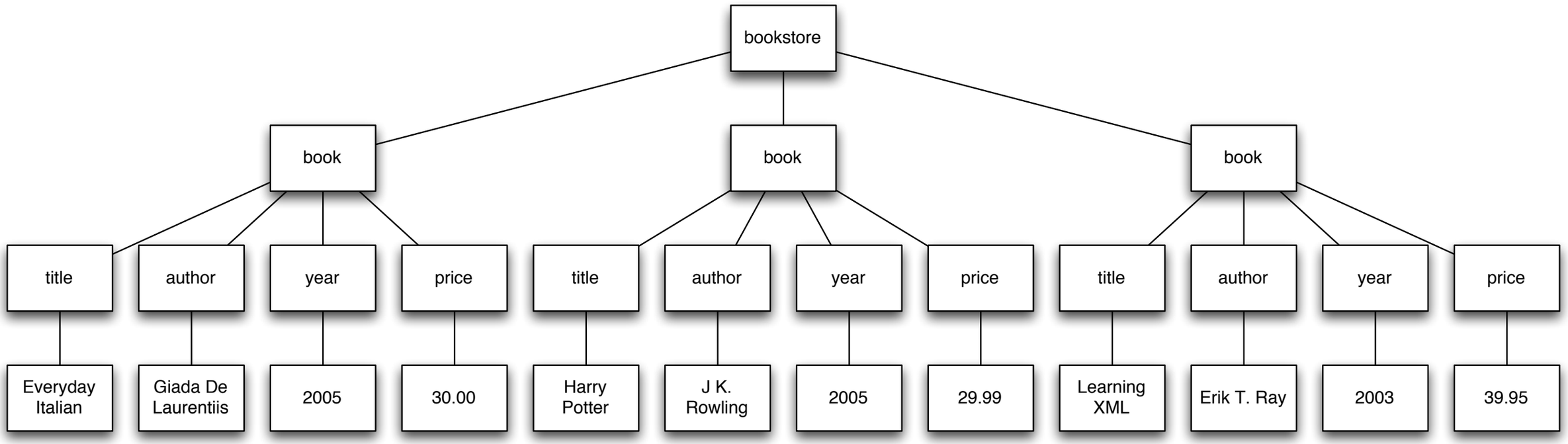
- XML is Not a Replacement for HTML
  - XML complements HTML
    - XHTML is an XML syntax compliant version of HTML
      - It has tags defined by a standards body

- XML Separates Data from HTML
- XML Simplifies Data Sharing
- XML Simplifies Data Transport
- XML Simplifies Platform Changes
- XML Makes Your Data More Available

- XML is Used to Create New Internet Languages
  - XHTML the latest version of HTML
  - WSDL for describing available web services
  - WAP and WML as markup languages for handheld devices
  - RSS languages for news feeds
  - RDF and OWL for describing resources and ontology
  - SMIL for describing multimedia for the web

- XML uses a tree structure
  - with a root element
  - and child elements
- tags indicate the start and end of an element
- opening tag looks like this:
  - <tag>
- a closing tag looks like this:
  - </tag>
- A valid XML document has exactly one closing tag for every opening tag


bookstore
book
book
book
title
author
year
price
title
author
year
price
title
author
year
price
Everyday Italian
Giada De Laurentiis
2005
30.00
Harry Potter
J K. Rowling
2005
29.99
Learning XML
Erik T. Ray
2003
39.95




```
<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>
```

Schema

Tags

Characters

```
<!DOCTYPE bookstore [

<!ELEMENT bookstore (book+)>
<!ELEMENT book (title,author,year,(price)+)>
<!ELEMENT title (CDATA)>
<!ELEMENT author (CDATA)>
<!ELEMENT year (CDATA)>
<!ELEMENT price (CDATA)>

<!ATTLIST book category CDATA #REQUIRED>
<!ATTLIST title lang CDATA #IMPLIED>

]>
```

- Details
  - All XML Elements Must Have a Closing Tag
  - HTML
    - This is a paragraph
    - This is another paragraph
- XML
  - This is a paragraph
  - This is another paragraph

- Details
  - XML Tags are Case Sensitive
    - <Message>This is incorrect</message>
    - <message>This is correct</message>
    - <Message>This is correct</Message>

- Details
  - XML Elements Must be Properly Nested
    - HTML might have this
      - This text is bold and italic
  - Valid XML requires this:
    - This text is bold and italic

- Details
  - XML Documents Must Have a Root Element
    - This is the top-level tag
      - <root>
      - <child>
      - <subchild>.....</subchild>
      - </child>
      - </root>

- Details
  - XML Nodes may have attributes
  - Which describe the tag
  - XML Attribute Values Must be Quoted
    - Invalid:

```
<note date=12/11/2007>
  <to>Tove</to>
  <from>Jani</from>
</note>
```

- Valid:

```
<note date="12/11/2007">
  <to>Tove</to>
  <from>Jani</from>
</note>
```

- Details
  - XML Nodes may have attributes
  - Which describe the tag
  - XML Attribute Values Must be Quoted
    - Invalid:

```
<note date=12/11/2007>
  <to>Tove</to>
  <from>Jani</from>
</note>
```

- Valid:

```
<note date="12/11/2007">
  <to>Tove</to>
  <from>Jani</from>
</note>
```

- Details
  - Special characters:
    - If you put a “<“ in your data it will mess up XML parsing
      - <message>if salary < 1000 then</message>
    - So 5 characters are special
      - <, >, &, ‘ , “
      - < , > , & , ' , "
      - <message>if salary < 1000 then</message>

- Details
  - Comments in XML
    - <!-- This is a comment -->
  - White-space is preserved
    - <message>There is a lot of space</message>

- Attributes and Elements are pretty interchangeable

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

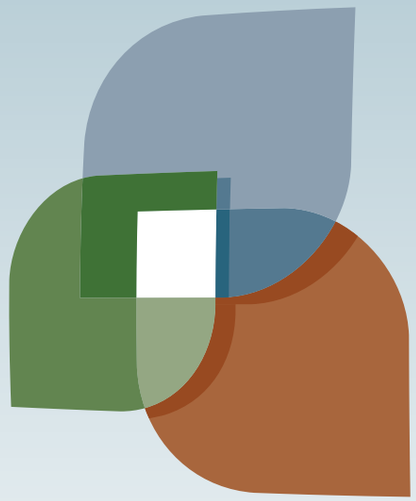
```
<person>
  <sex>female</sex>
  <firstname>Anna</firstname>
  <lastname>Smith</lastname>
</person>
```

```
<note date="10/01/2008">
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>

<note>
  <date>10/01/2008</date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>

<note>
  <date>
    <day>10</day>
    <month>01</month>
    <year>2008</year>
  </date>
  <to>Tove</to>
  <from>Jani</from>
  <heading>Reminder</heading>
  <body>Don't forget me this weekend!</body>
</note>
```

- On beyond XML
  - XML validation
  - Schemas like XML - DTD
  - Namespaces
- XSLT
  - transforms XML to HTML for viewing



LUCI