

XML “Elements” (or Nodes)

- Simple Element
- Complex Element

```
<people>
  <person>
    <name>Chuck</name>
    <phone>303 4456</phone>
  </person>
  <person>
    <name>Noah</name>
    <phone>622 7421</phone>
  </person>
</people>
```

eXtensible Markup Language

- Primary purpose is to help information systems **share structured data**
- It started as a simplified subset of the Standard Generalized Markup Language (SGML), and is designed to be relatively human-legible

<http://en.wikipedia.org/wiki/XML>

XML Basics

- Start Tag
- End Tag
- Text Content
- Attribute
- Self Closing Tag

```
<person>  
  <name>Chuck</name>  
  <phone type="intl">  
    +1 734 303 4456  
  </phone>  
  <email hide="yes" />  
</person>
```

White Space

```
<person>  
  <name>Chuck</name>  
  <phone type="intl">  
    +1 734 303 4456  
  </phone>  
  <email hide="yes" />  
</person>
```

Line ends do not matter.
White space is generally
discarded on text elements.
We indent only to be
readable.

```
<person>  
  <name>Chuck</name>  
  <phone type="intl">+1 734 303 4456</phone>  
  <email hide="yes" />  
</person>
```

Some XML...

```
<recipe name="bread" prep_time="5 mins" cook_time="3 hours">
  <title>Basic bread</title>
  <ingredient amount="8" unit="dL">Flour</ingredient>
  <ingredient amount="10" unit="grams">Yeast</ingredient>
  <ingredient amount="4" unit="dL" state="warm">Water</ingredient>
  <ingredient amount="1" unit="teaspoon">Salt</ingredient>
  <instructions>
    <step>Mix all ingredients together.</step>
    <step>Knead thoroughly.</step>
    <step>Cover with a cloth, and leave for one hour in warm room.</step>
    <step>Knead again.</step>
    <step>Place in a bread baking tin.</step>
    <step>Cover with a cloth, and leave for one hour in warm room.</step>
    <step>Bake in the oven at 180(degrees)C for 30 minutes.</step>
  </instructions>
</recipe>
```

<http://en.wikipedia.org/wiki/XML>

XML Terminology

- **Tags** indicate the beginning and ending of elements
- **Attributes** - Keyword/value pairs on the opening tag of XML
- **Serialize / De-Serialize** - Convert data in one program into a common format that can be stored and/or transmitted between systems in a programming language-independent manner

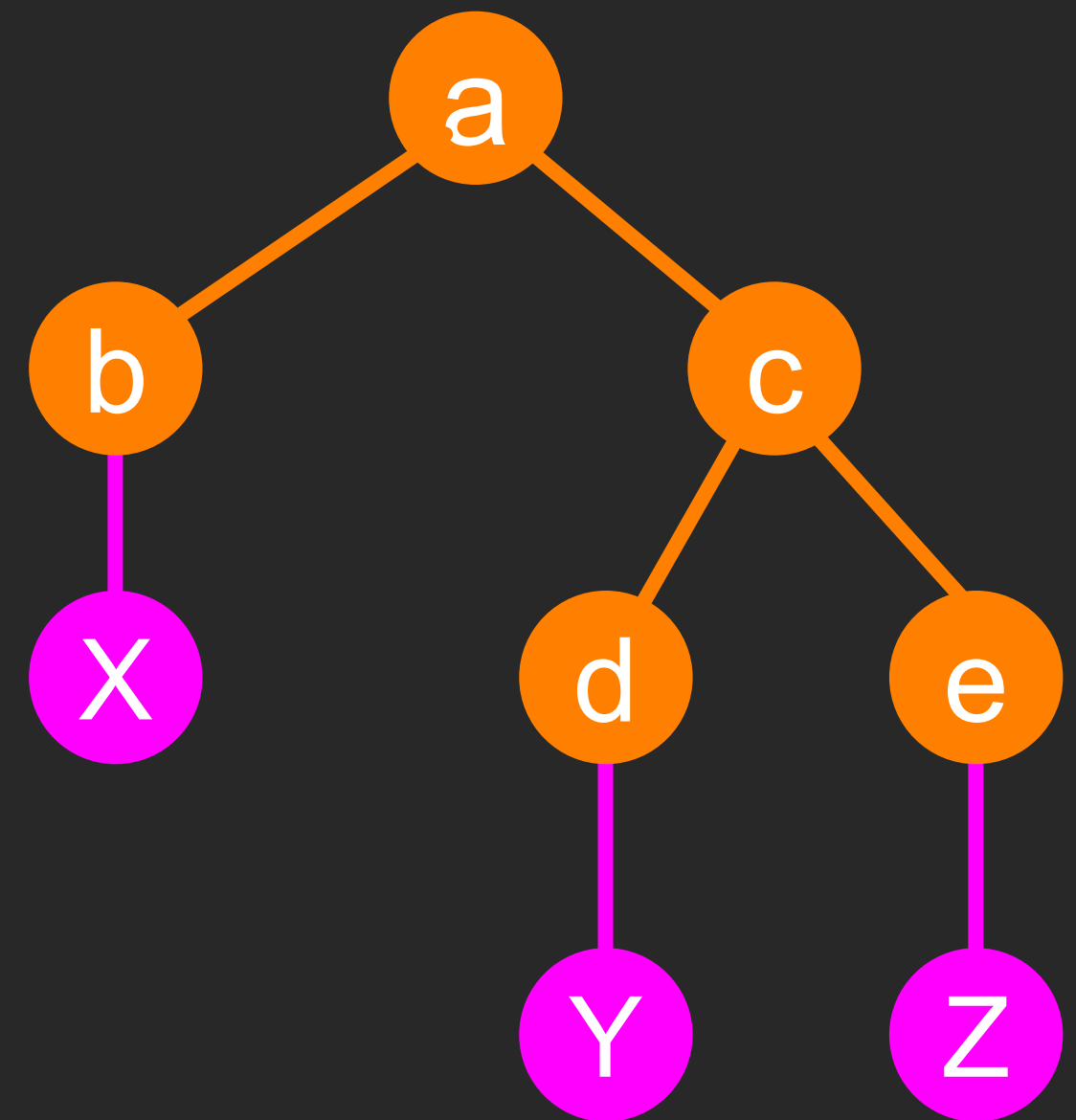
<http://en.wikipedia.org/wiki/Serialization>

XML as a Tree

```
<a>  
  <b>X</b>  
  <c>  
    <d>Y</d>  
    <e>Z</e>  
  </c>  
</a>
```

Elements

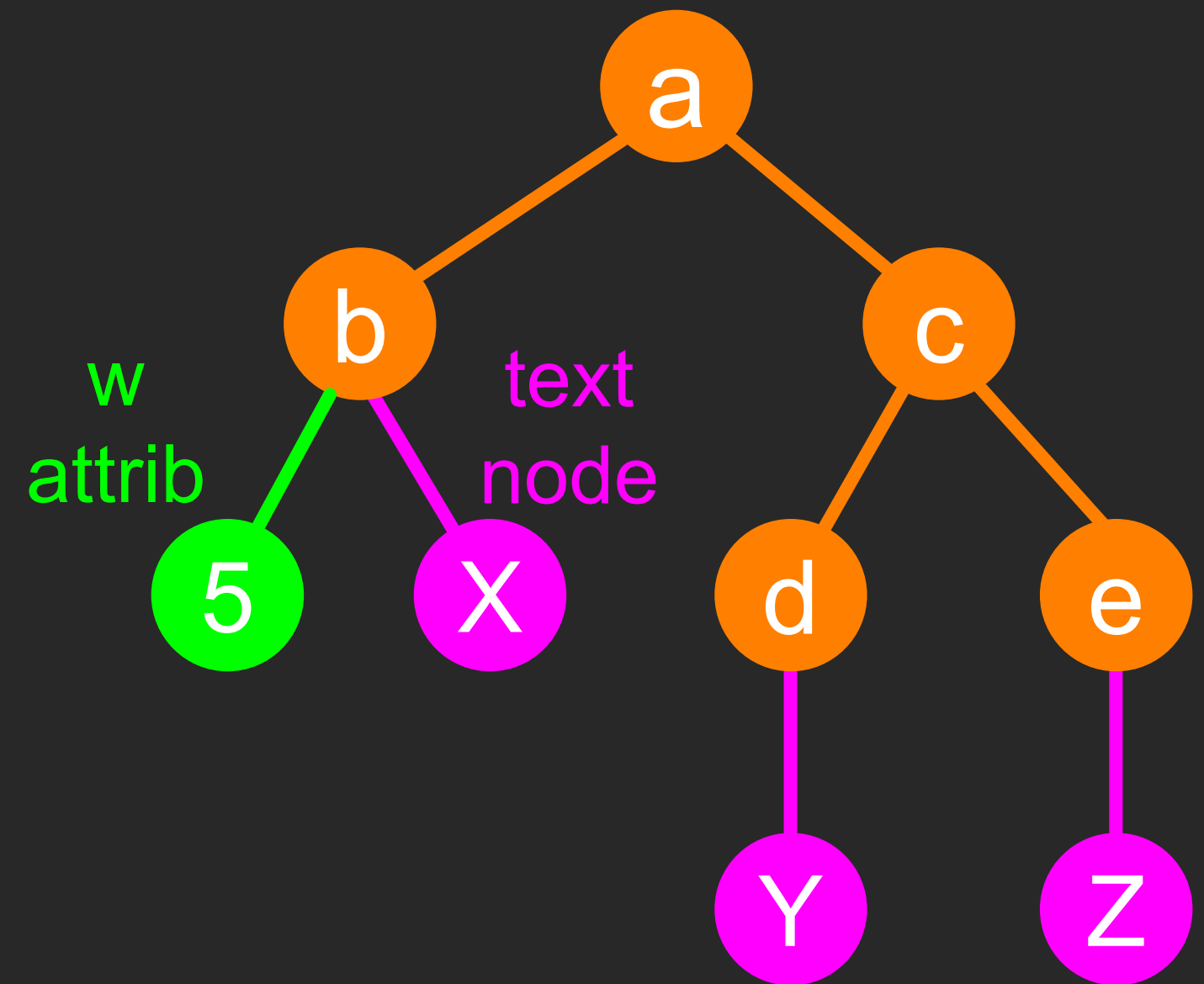
Text



XML Text and Attributes

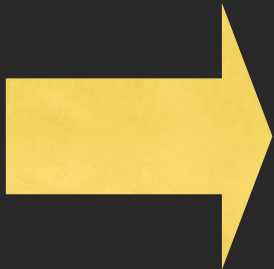
```
<a>  
  <b w="5">X</b>  
  <c>  
    <d>Y</d>  
    <e>Z</e>  
  </c>  
</a>
```

Elements Text



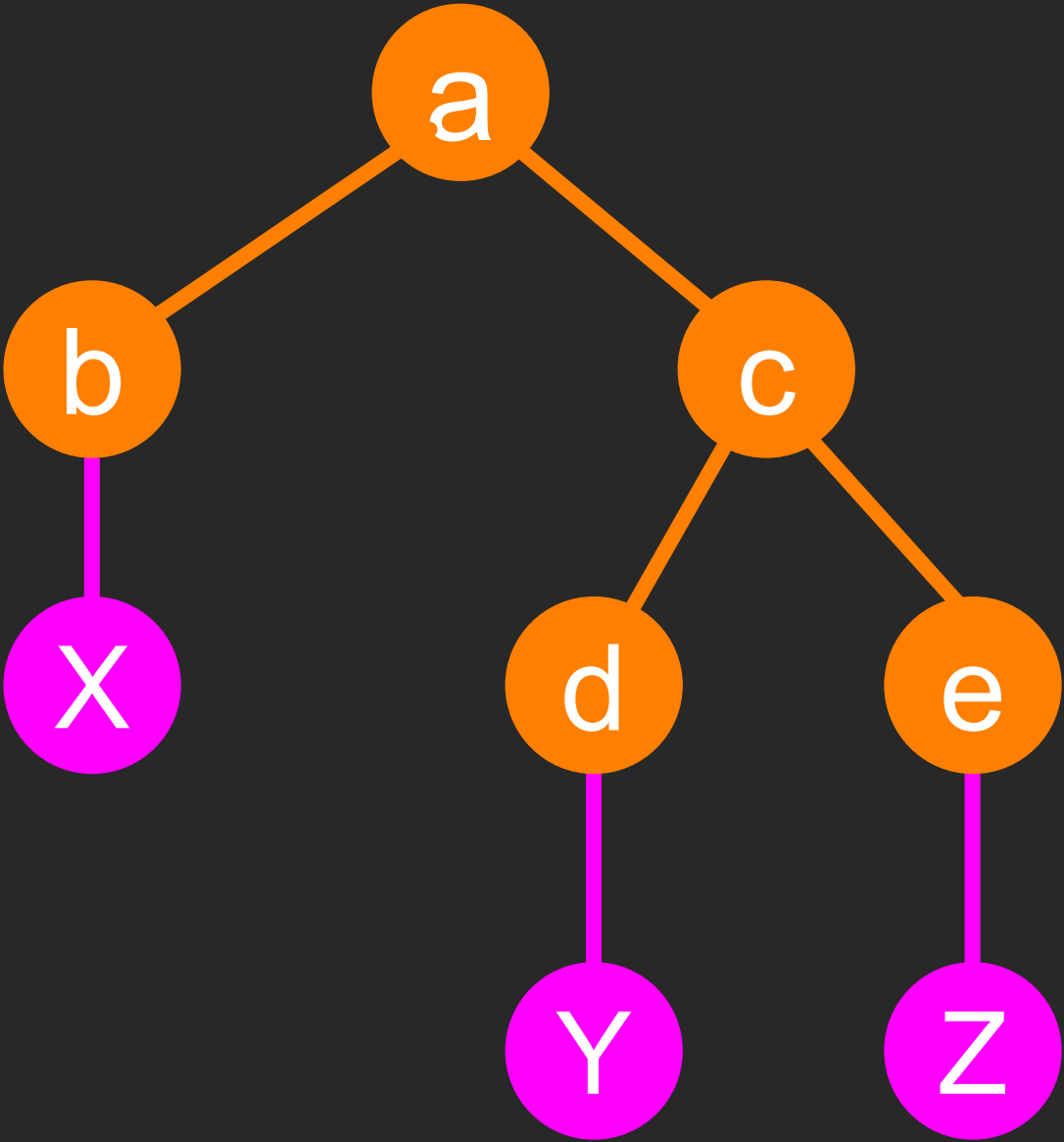
XML as Paths

```
<a>  
  <b>X</b>  
  <c>  
    <d>Y</d>  
    <e>Z</e>  
  </c>  
</a>
```



| | |
|--------|---|
| /a/b | X |
| /a/c/d | Y |
| /a/c/e | Z |

Elements Text



XML Schema

Describing a “**contract**” as to what is acceptable XML.

http://en.wikipedia.org/wiki/Xml_schema

http://en.wikibooks.org/wiki/XML_Schema



Acknowledgements / Contributions



These slides are Copyright 2010- Charles R. Severance (www.dr-chuck.com) of the University of Michigan School of Information and open.umich.edu and made available under a Creative Commons Attribution 4.0 License. Please maintain this last slide in all copies of the document to comply with the attribution requirements of the license. If you make a change, feel free to add your name and organization to the list of contributors on this page as you republish the materials.

Initial Development: Charles Severance, University of Michigan School of Information

... Insert new Contributors here