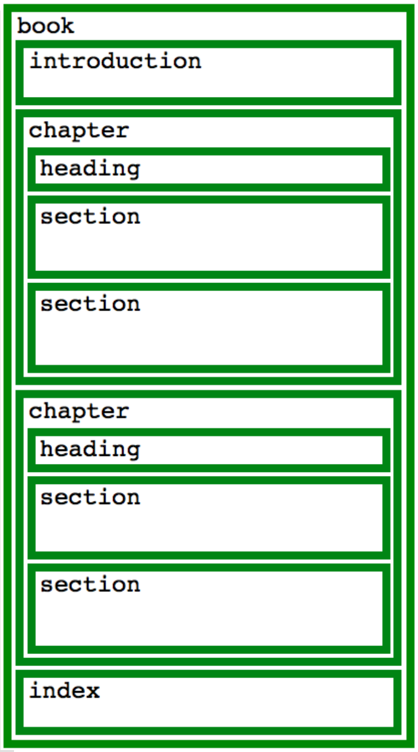
Navigating the XML Tree

Syd Bauman

XML representations

```
<?xml version="1.0"
 encoding="UTF-8"?>
<book>
  <introduction>Blah blah
 blah ... </introduction>
  <chapter>
    <heading>Wines</
heading>
    <section>White
 wines ... </section>
    <section>Red wines ...
 </section>
  </chapter>
  <chapter>
    <heading>Beers</
heading>
    <section>Ales ... </
section>
    <section>Lagers ... </
section>
  </chapter>
  <index> stuff ... </
index>
</book>
```



book

introduction

chapter

heading

section

section

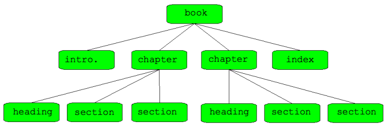
chapter

heading

section

section

index



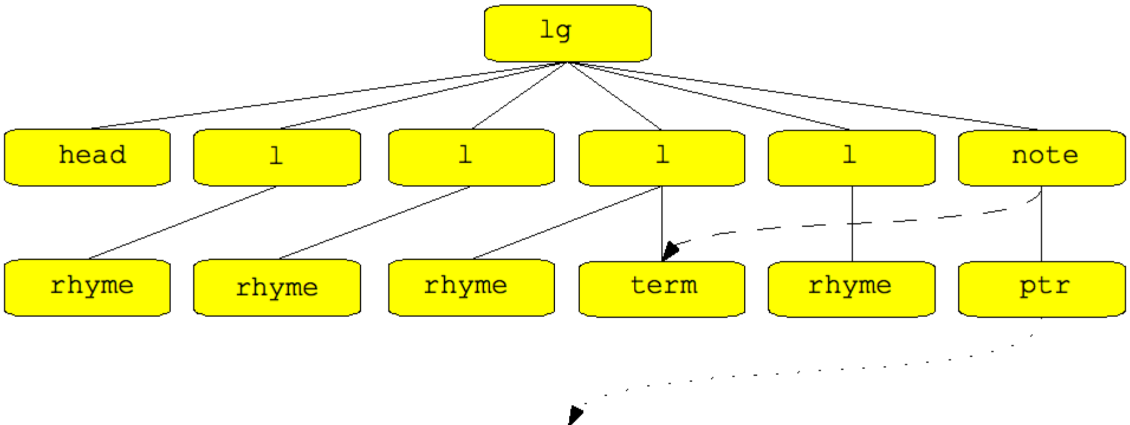
Note that pressing Control-minus and Control-plus will make the text larger or smaller, allowing the images to grow or shrink too.

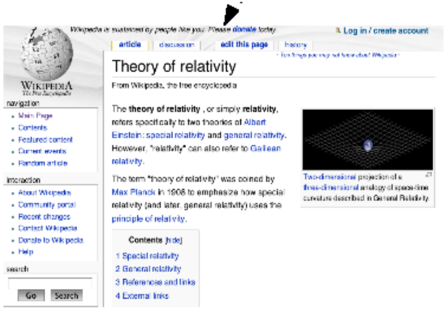
Sample document fragment

```
<?xml version="1.0" encoding="UTF-8"?>
<lg type="limerick" rhyme="aabba" n="3">
  <head>Warp Speed, Ms Bright!</head>
  <l>There was a young lady named <rhyme label="a">Bright</rhyme>,</l>
  <l>Who travelled much faster than <rhyme label="a">light</rhyme>,</l>
  <l>She departed one <rhyme label="b">day</rhyme>,</l>
  <l>In a <term xml:id="t17">relative</term> way <rhyme label="b">way</
rhyme>,</l>
  <l>And returned on the previous <rhyme label="a">night</rhyme>.</l>
  <note target="#t17">See
    <ptr target="http://en.wikipedia.org/wiki/Theory_of_relativity"/>.</note>
</lg>
```

Sample portion of TEI tree

Grossly simplified sub-tree representation of a TEI document.




Wikipedia is sustained by people like you. Please donate *today.*

Log in / create account

article | discussion | edit this page | history

· Ten things you may not know about Wikipedia ·

Theory of relativity

From Wikipedia, the free encyclopedia

The **theory of relativity** , or simply **relativity**, refers specifically to two theories of Albert Einstein: special relativity and general relativity. However, "relativity" can also refer to Galilean relativity.

The term "theory of relativity" was coined by Max Planck in 1906 to emphasize how special relativity (and later, general relativity) uses the principle of relativity.

Two-dimensional projection of a three-dimensional analogy of space-time curvature described in General Relativity.

Contents [hide]

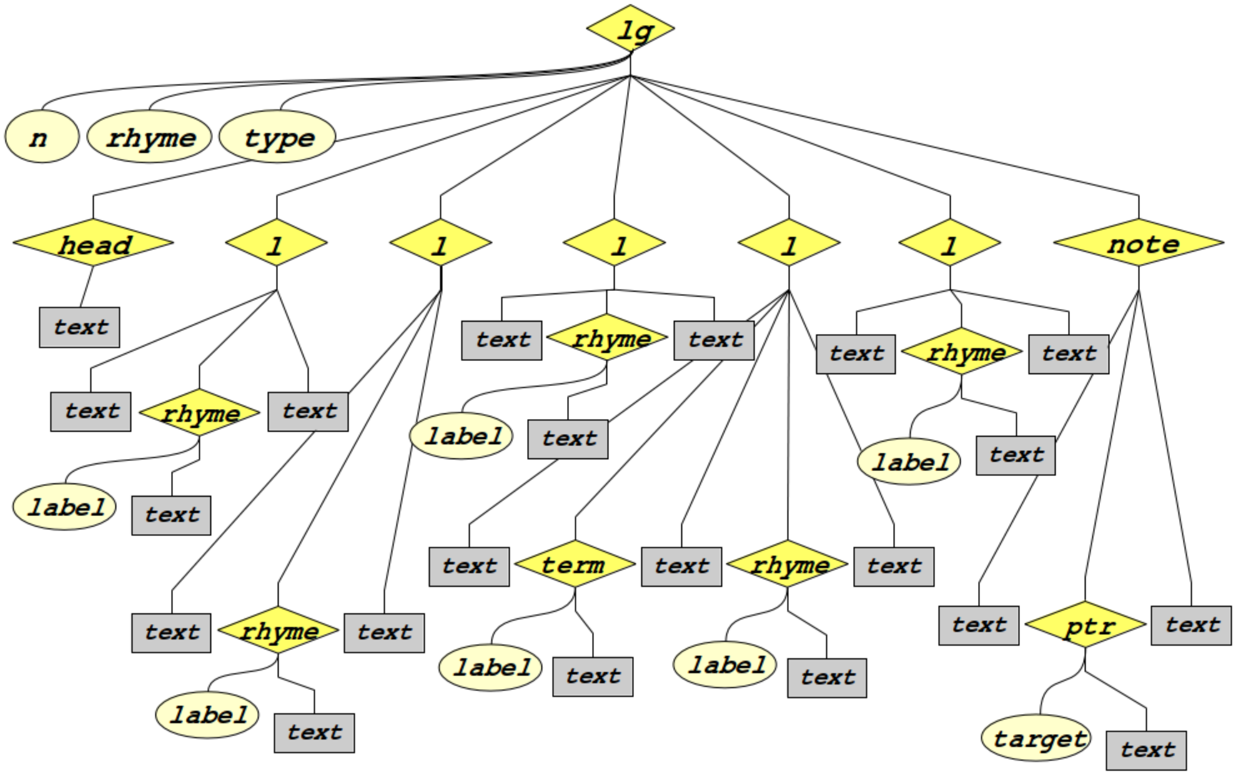
1 Special relativity
2 General relativity
3 References and links
4 External links

There was a young lady named Bright, Whose speed was far faster than light; She started one day In a relative way, sAnd returned on the previous night.

A. H. Reginald Buller in Punch (Dec. 19, 1923): 591.

XSLT view

Same fragment, but adding text and attribute nodes

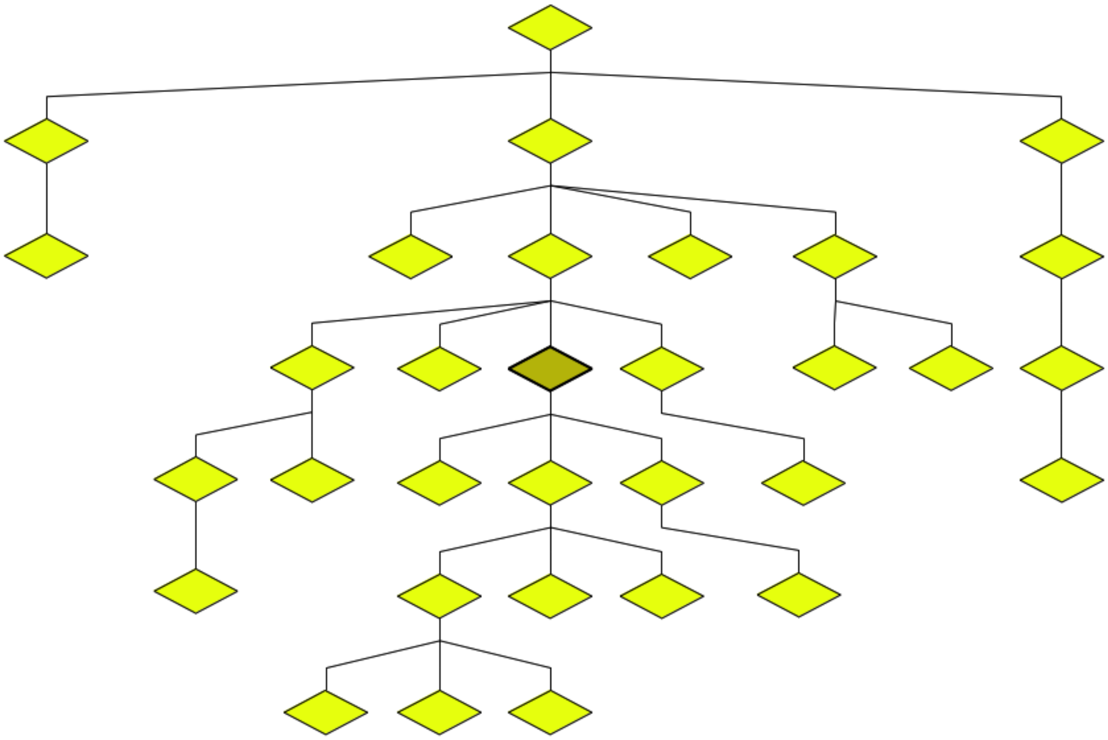


Element nodes have just the element's local name; attribute nodes have just the name of the attribute; text nodes have just the word text.

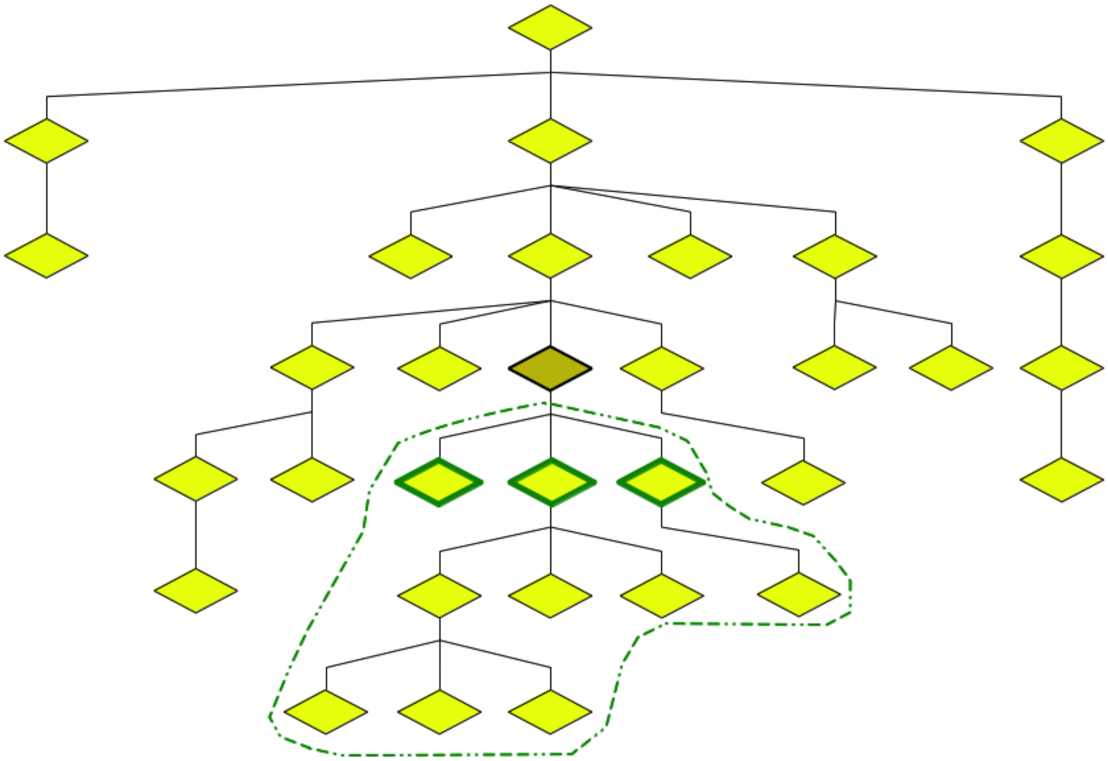
The context node

- Important definition: The **context node** is where we are now in the XML tree.
- In XPath, and therefore in XSLT, the processor is always somewhere in the tree.
- From the context node, you can travel anywhere else in the tree.
- We do that by travelling along **XPath axes**.

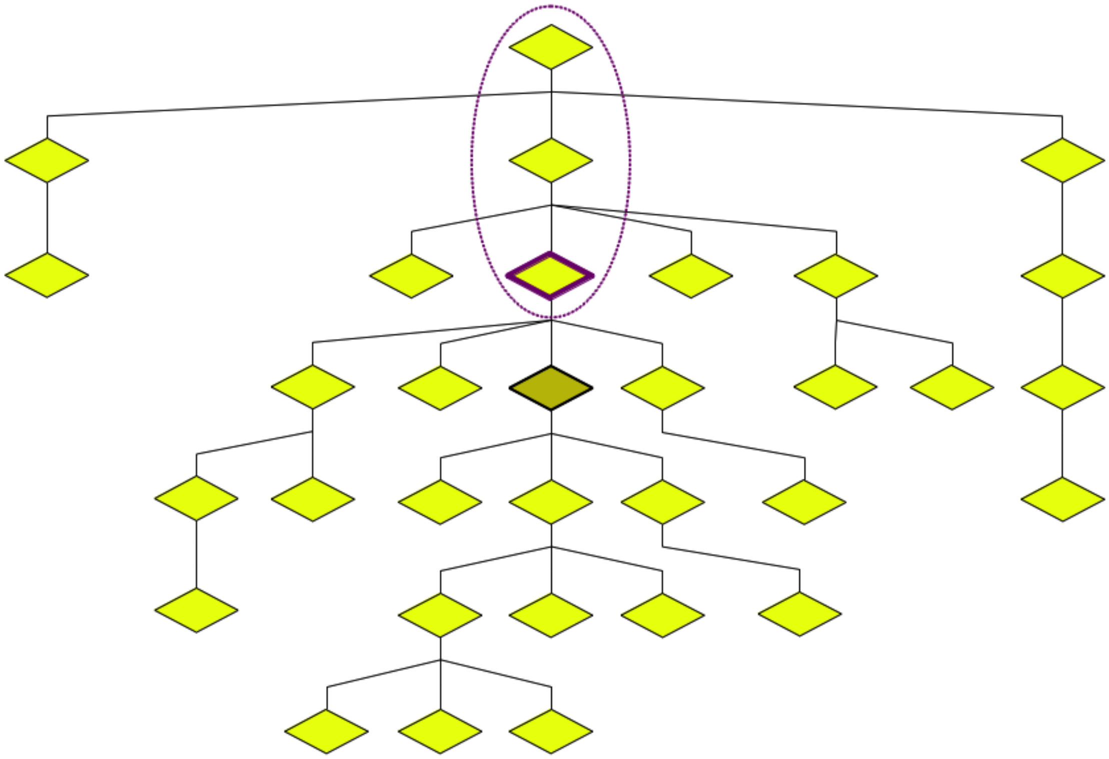
Axes: `self::`



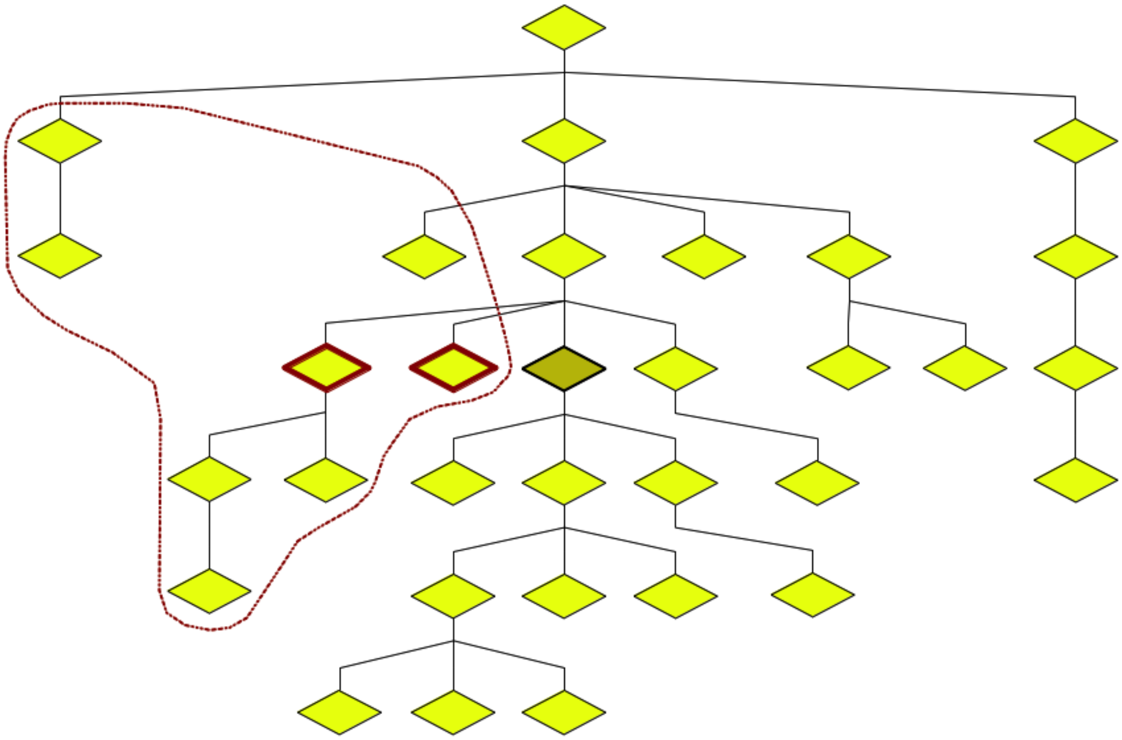
Descendants (`child::` and `descendant::`)



Ancestors (`parent::` and `ancestor::`)

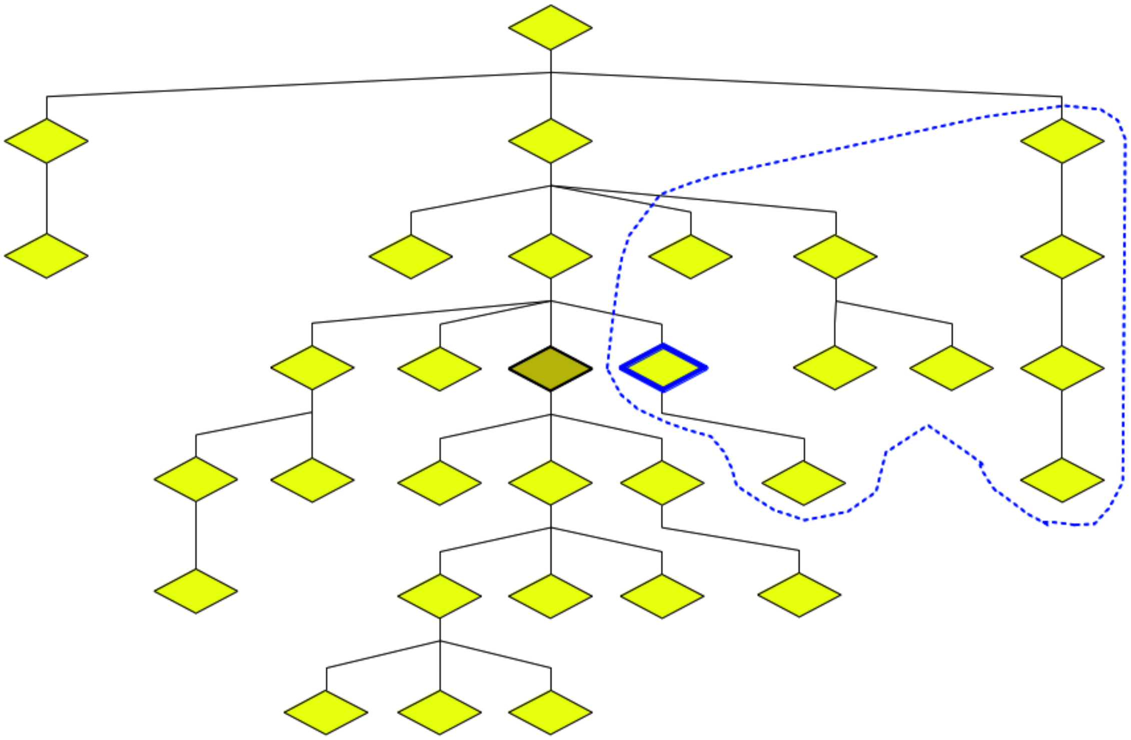


Preceding (`preceding::` and `preceding-sibling::`)



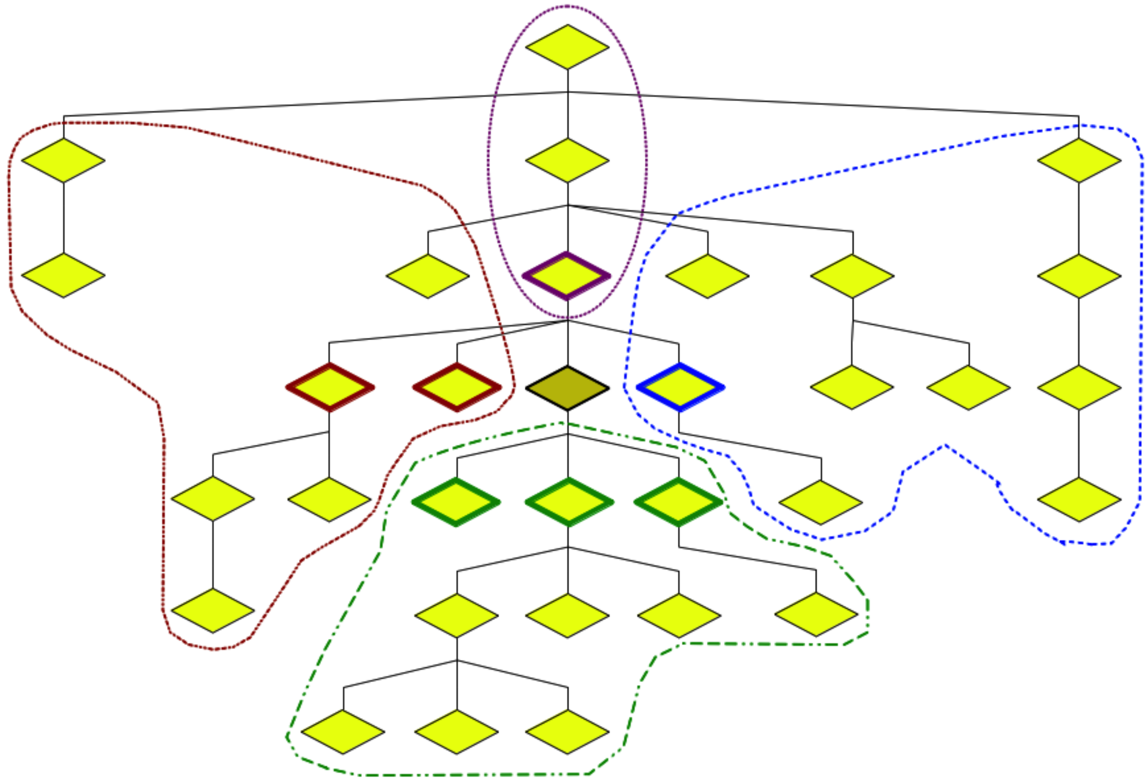
This is where the analogy with genealogical trees breaks down. The preceding:: axis includes a set of nodes which do not match any familial group. It's basically all the nodes which both started and finished before the context node.

following (`following::` and `following-sibling::`)



The following axis contains only elements which start after the context node has finished.

Axes: the "XPath Butterfly"



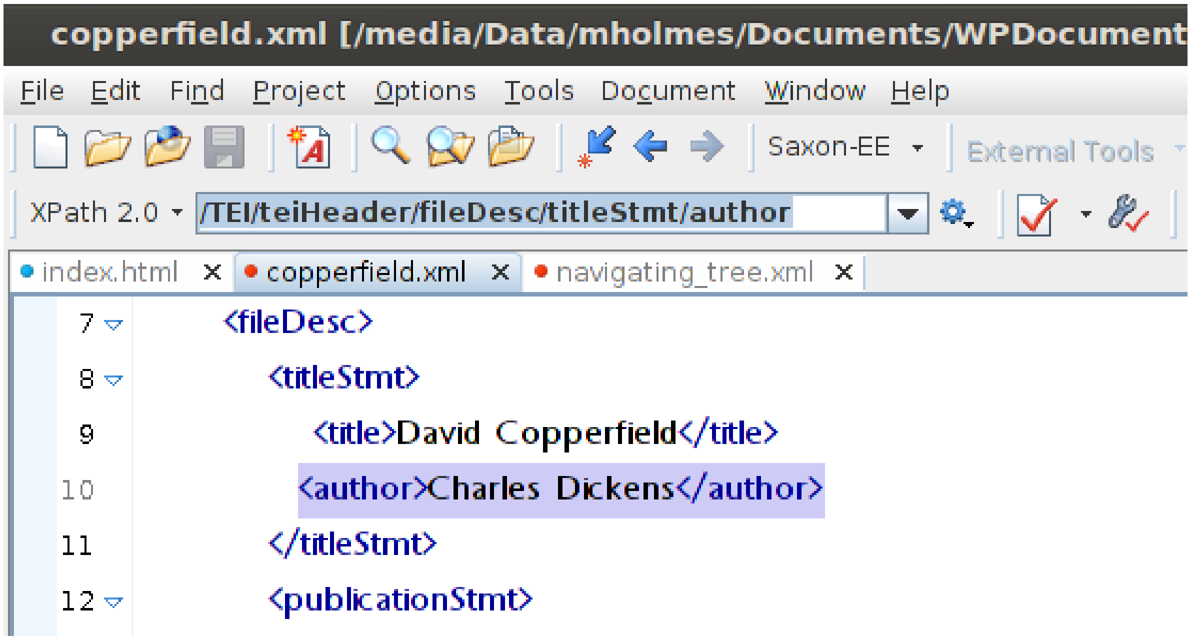
XPath …

… lets us (among other things) select nodes in the tree.

… is used by XSLT, XQuery, Schematron; XLink and XPointer, too.

Try it!

- Download the example short David Copperfield XML file and open it in Oxygen: http://web.uvic.ca/~mholmes/dhoxss2013/examples/copperfield.xml
- In the upper left corner there is a text-entry field (looks like a search box). Ensure the box is labelled XPath 2.0. Then type in `/TEI/teiHeader/fileDesc/titleStmt/author`.



XPath 2.0 /TEI/teiHeader/fileDesc/titleStmt/author

```
<fileDesc>
  <titleStmt>
    <title>David Copperfield</title>
    <author>Charles Dickens</author>
  </titleStmt>
  <publicationStmt>
```

Basic filepath-like path expressions

A bare-bones XPath expression is similar to filesystem addressing: if the path starts with a solidus (/ aka "forward slash"), then it represents a path from the root; if it does not start with a solidus then it represents a path from "here".

```
/TEI/teiHeader/fileDesc/titleStmt/title
```

```
list/item/label
```

What's another way of saying "here"?

XPath axes: me, myself, I

short	long	means	try it!
`.`	`self::node()`	me, whatever I happen to be (element, attribute, comment, processing instruction, document root, or even text)	oXygen shows node cursor is in

(Kay, chapter 9)

XPath axes: my children

short	long	means	try it!
`head`	`child::head`	my `<head>` children	List all `<head>` elements for `<div>` children of the body
`*`	`child::*`	my **element** children, whatever they are	List all elements which are children of `<div>`.

(Kay, chapter 9)

`/TEI/text/body/div/head`, or `/TEI/text/body/div/child::head`

XPath axes: my descendants

short	long	means	try it!
`//div`	`descendant::div`	my `<div>` descendants (note: long and short are not exactly equivalent)	List all the `<head>` elements

(Kay, chapter 9)

`/TEI/descendant::head` but compare `//head` with `/TEI/text/body//head`; even when the cursor is in the `<body>` tag, the `//` takes us back to the root.

XPath axes: my parent

short	long	means	try it!
`..`	`parent::node()`	my parent, whatever it happens to be (element or root)	List the parents of `<note>` elements
[none]	`parent::author`	my parent, if it is an `<author>`	List the `<div>` parents of `<p>` elements

(Kay, chapter 9)

Contrast the use of `//p/parent::div` with `//p/parent::node()`.

XPath axes: my ancestors

short	long	means	try it!
[none]	`ancestor::*`	all my ancestors	List the ancestors of the `<sourceDesc>` element

(Kay, chapter 9)

`Note that the ancestor axis ultimately includes everything.`

XPath axes: following and following-sibling

short	long	means	try it!
[none]	`following::p`	all the `<p>`s that come after me	Find all the `<p>` elements that occur **after** the `<publicationStmt>`
[none]	`following-sibling::*`	the children of my parent that come after me	Find all the children of `<div>` that follow `<head>`

(Kay, chapter 9)

1) `/TEI/teiHeader/fileDesc/publicationStmt/following::p` or `//publicationStmt/following::p`. Note how it **doesn't** select the `<p>` child of `<publicationStmt>`.

2) `//head/following-sibling::*`. Note that `<note>` elements are not included -- why?

XPath axes: preceding and preceding-sibling

short	long	means	try it!
[none]	`preceding::p`	the `<p>`s that come before me	put your cursor in the last `<p>` tag in the document, and find all the preceding `<p>` tags.
[none]	`preceding-sibling::p`	the `<p>` children of my parent that come before me	How many preceding-sibling `<p>` tags are there for the last `<p>` tag in the document?

(Kay, chapter 9)

XPath axes: attributes

short	long	means	try it!
`@target`	`attribute::target`	my `@target` attribute	list all the `@target` attributes
`@*`	`attribute::*`	all of my attributes	How many attributes are there in the document?

(Kay, chapter 9)

Note that the attributes **don't** include the `xmlns` declaration.

XPath axes: summary

- self::
- child::, descendant::
- parent::, ancestor::
- following::, following-sibling::
- preceding::, preceding-sibling::
- attribute::