XML Data Management

5. Extracting Data from XML: XPath

Werner Nutt

based on slides by Sara Cohen, Jerusalem

Extracting Data from XML

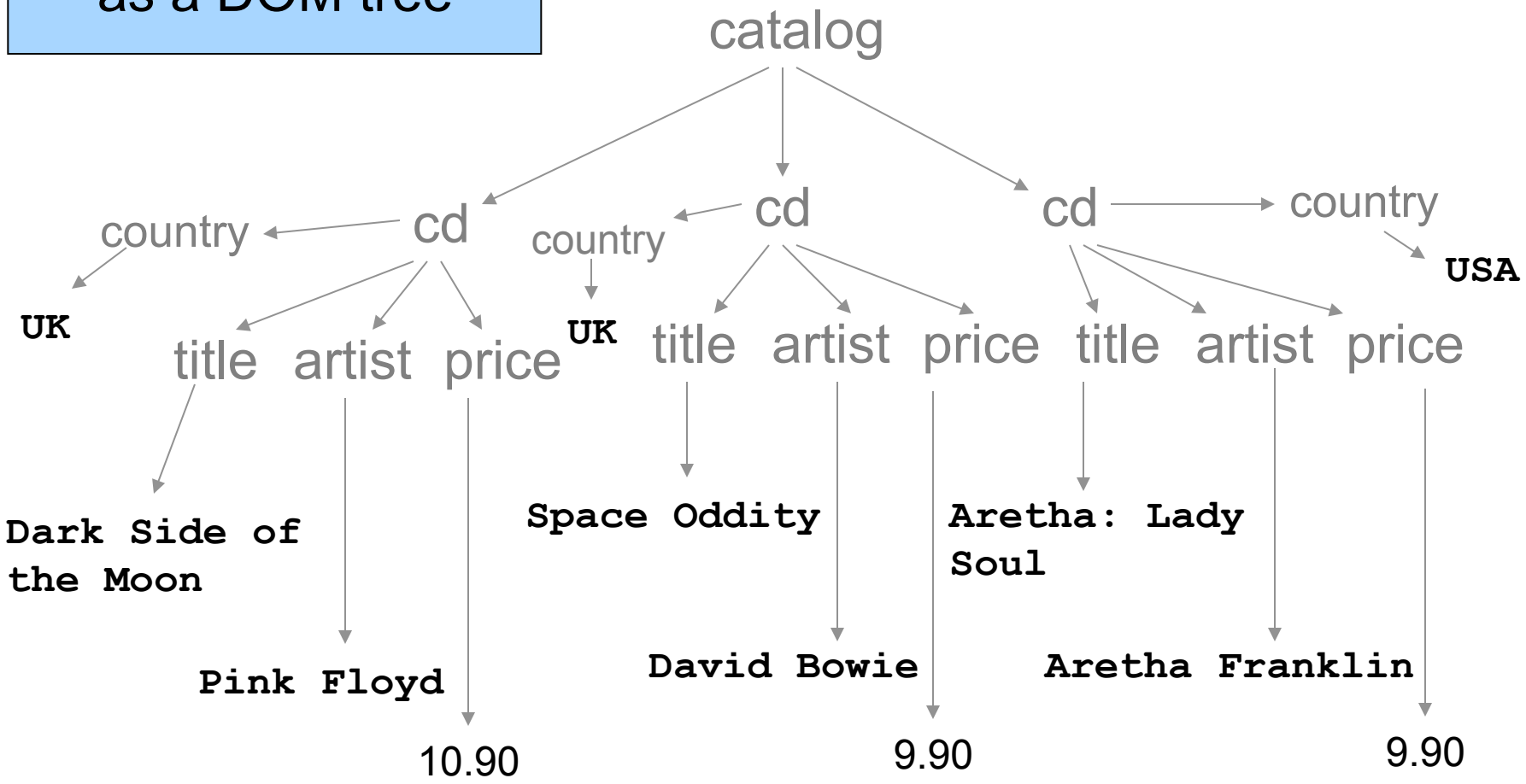
- Data stored in an XML document must be extracted to use it with various applications
- Data can be extracted by a *program …*
- … or using a *declarative* language: XPath
- XPath is used extensively in other languages, e.g.,
  - XSL
  - XML Schema
  - XQuery
  - Xpointer
- Versions: XPath 1.0 (allows for efficient execution),
  XPath 2.0 (not yet widely supported)

```
<?xml version="1.0" encoding="ISO-8859-1"?>
<catalog>
    <cd country="UK">
      <title>Dark Side of the Moon</title>
      <artist>Pink Floyd</artist>
      <price>10.90</price>
    </cd>
    <cd country="UK">
        <title>Space Oddity</title>
        <artist>David Bowie</artist>
        <price>9.90</price>
    </cd>
    <cd country="USA">
        <title>Aretha: Lady Soul</title>
        <artist>Aretha Franklin</artist>
        <price>9.90</price>
   </cd>
</catalog>
```

Our XML document

The XML document as a DOM tree

catalog.xml



XPath: Ideas

A language of path expressions:

- a document *D* is a tree
- an expression *E* specifies possible paths in *D*
- *E* returns nodes in *D* that can be reached from the root walking along an *E*-path

Path expressions specify

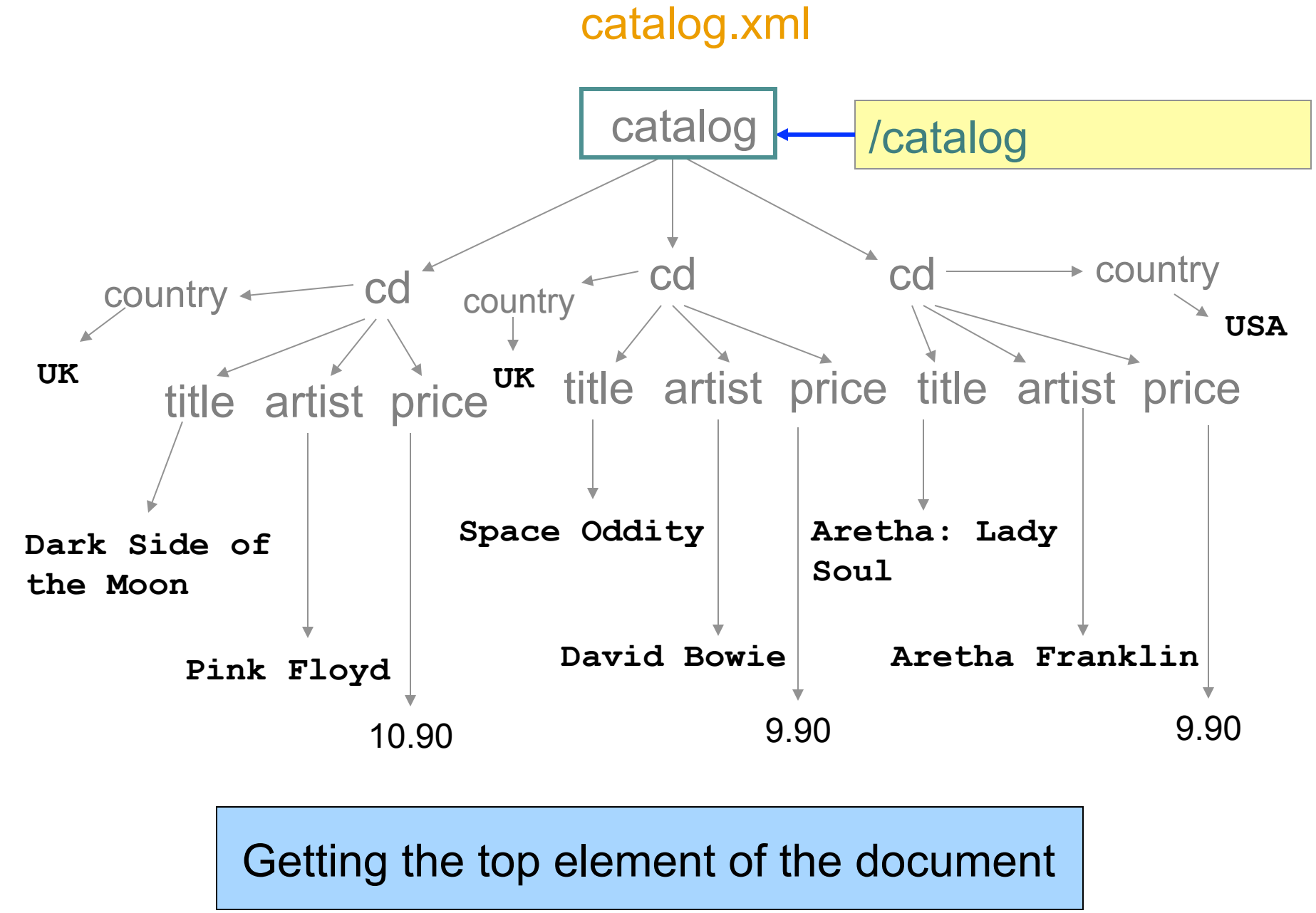
- navigation in docs
- tests on nodes

XPath Syntax: Path Expressions

- / at the beginning of an XPath expression
  represents the root of the document
- / between element names
  represents a parent-child relationship
- // represents an ancestor-descendant relationship
- foo element name, path has to go through an element foo,
  e.g., /cd
- * wildcard, represents any element
- @ marks an attribute

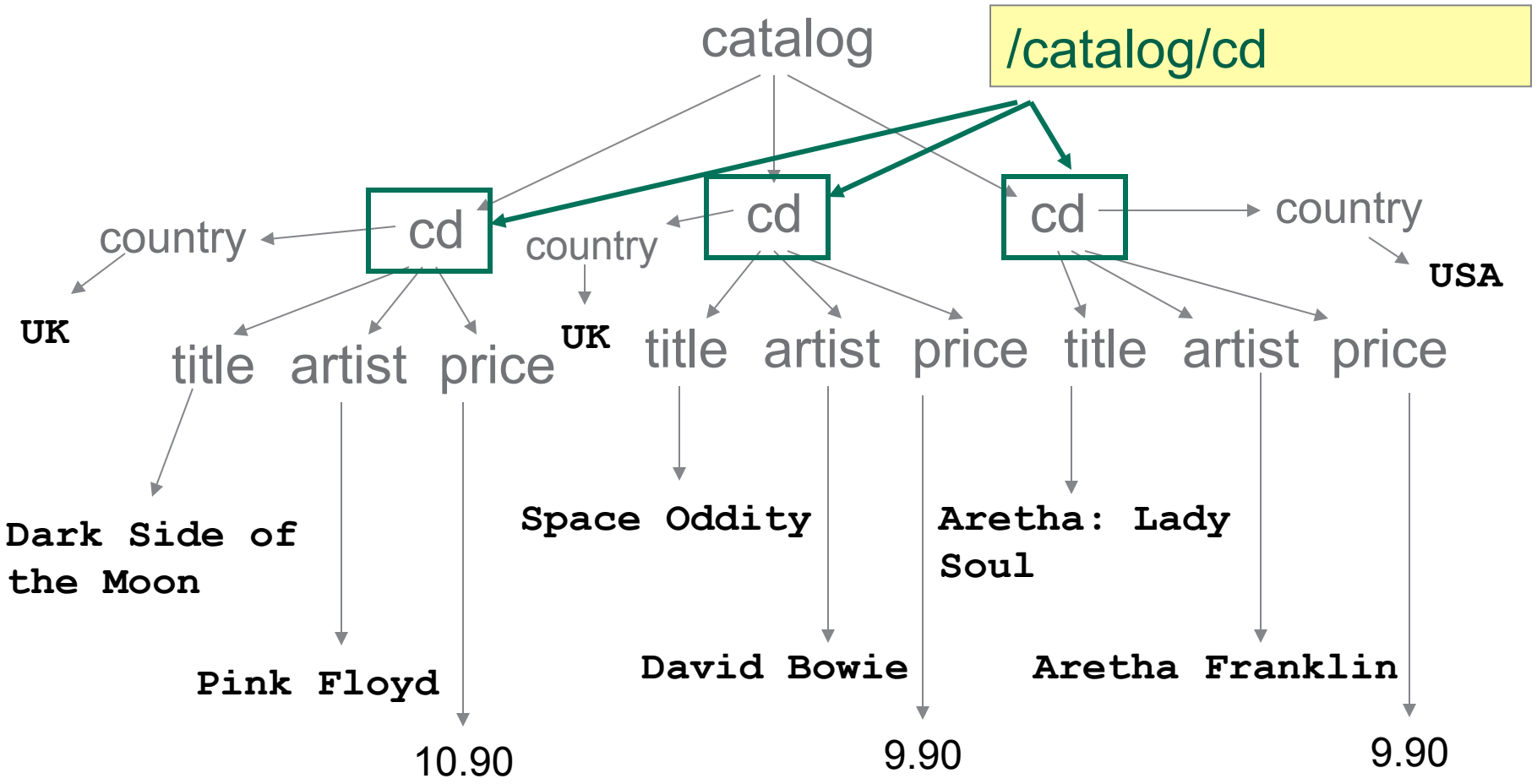
XPath Syntax: Conditions and Built-Ins

- [*condition*] specifies a condition, e.g., /cd[price < 10]
- [N] position of a child, e.g., /cd[2]
- contains(s1,s2) string comparison, e.g.,
  /cd[contains(title, "Moon")]
- name() name of an element, e.g., /*[name()="cd"]
  is equivalent to /cd



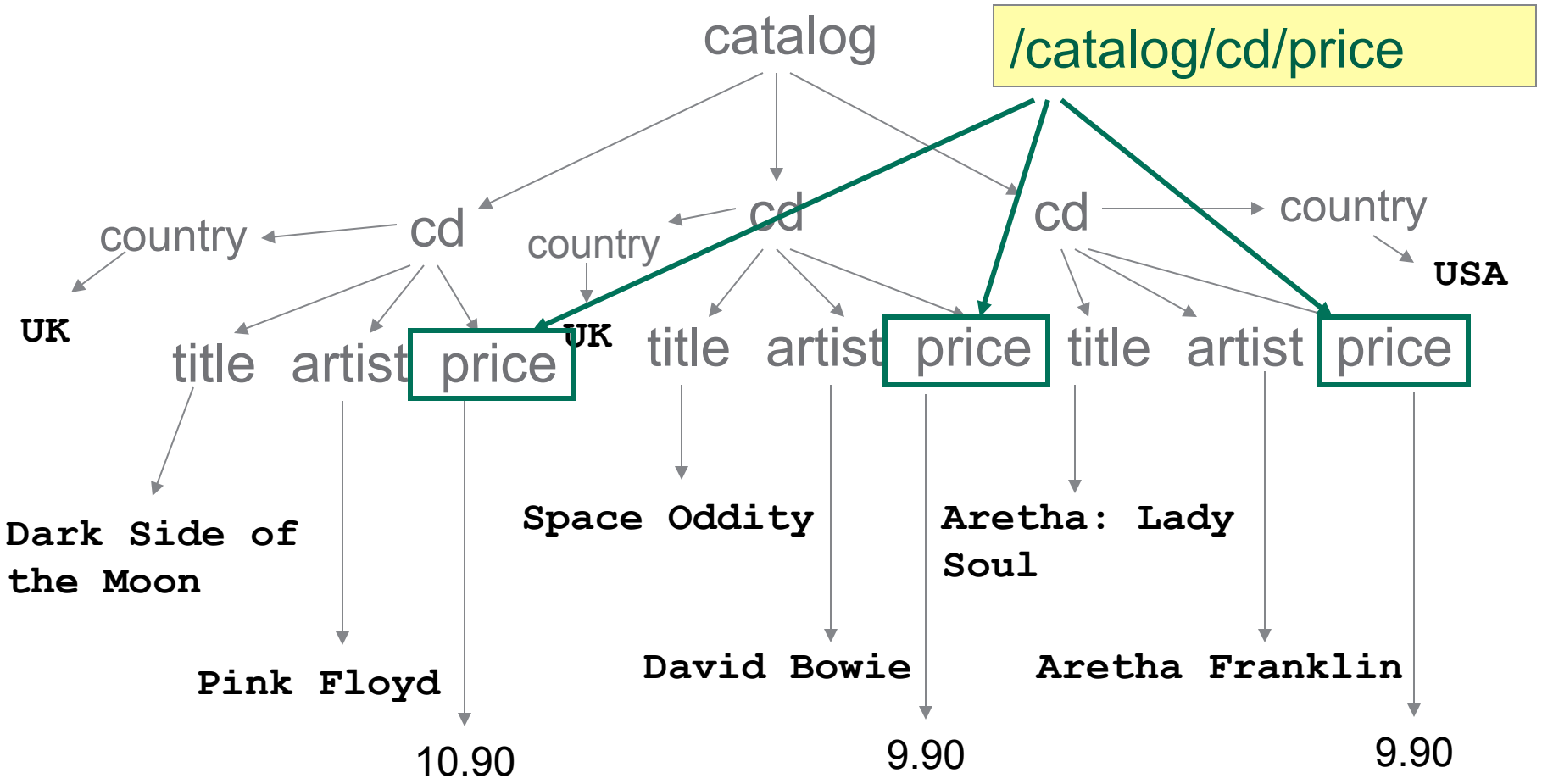
Getting the top element of the document

catalog.xml



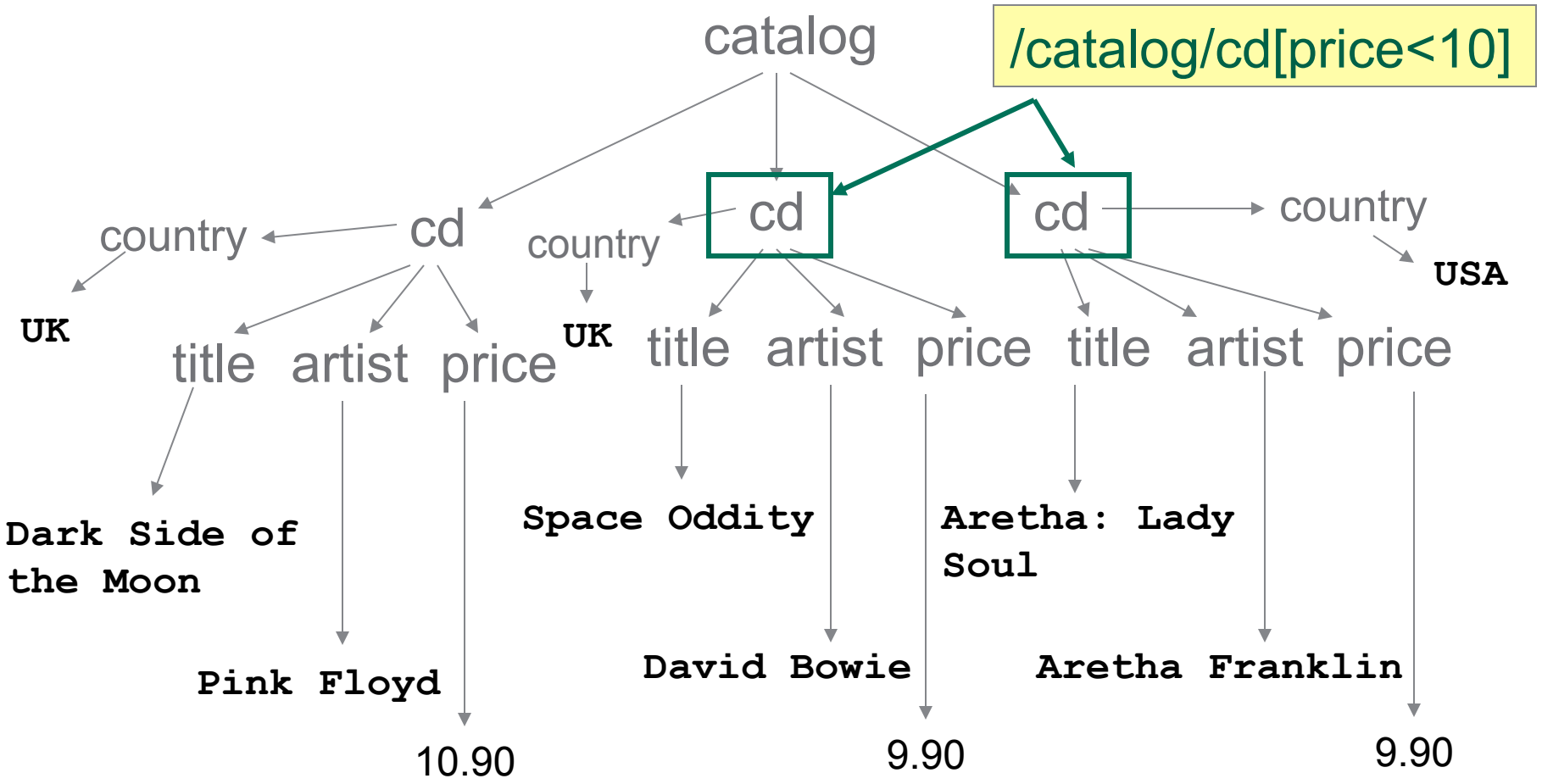
Finding child nodes

catalog.xml

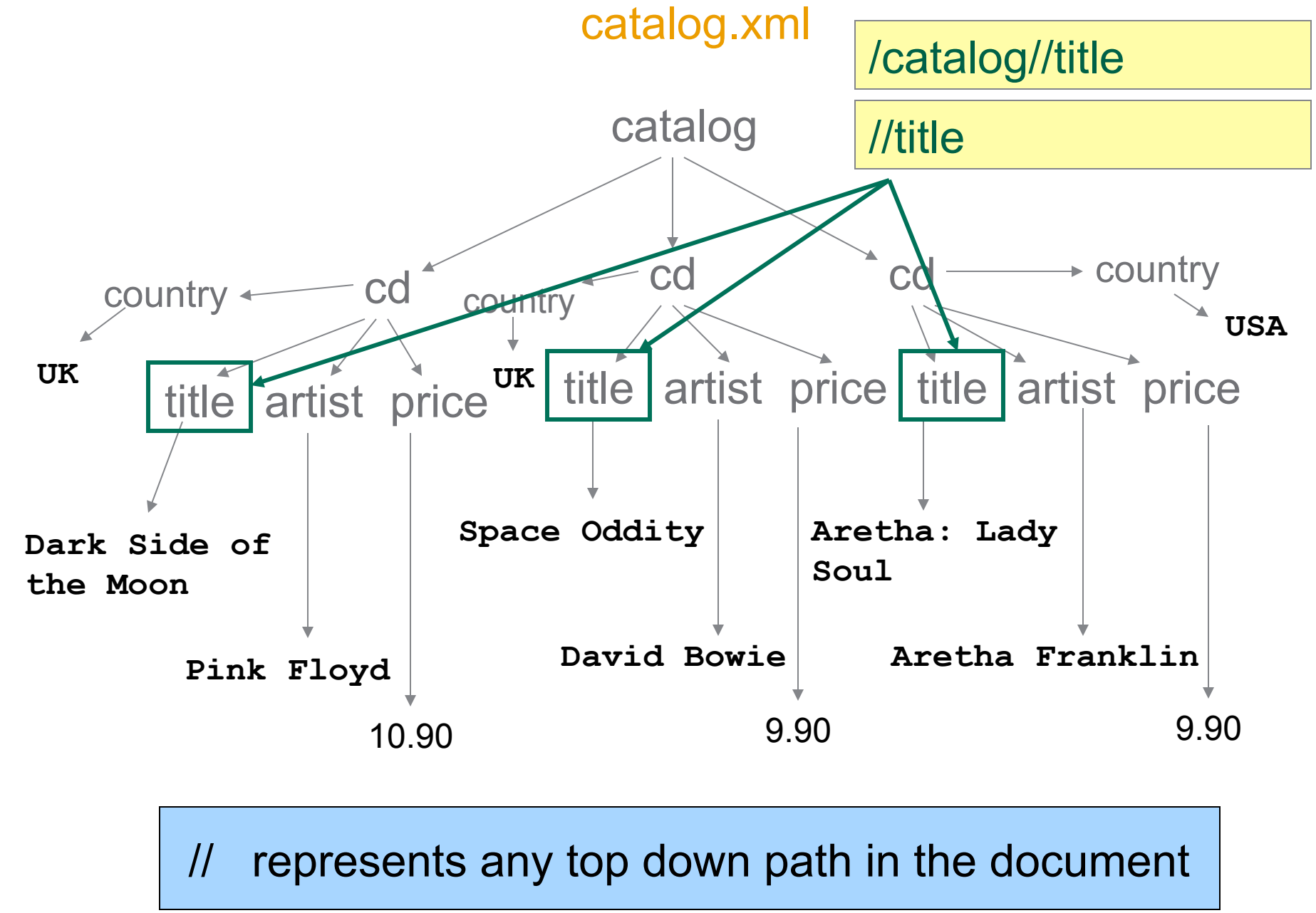


Finding descendant nodes

catalog.xml

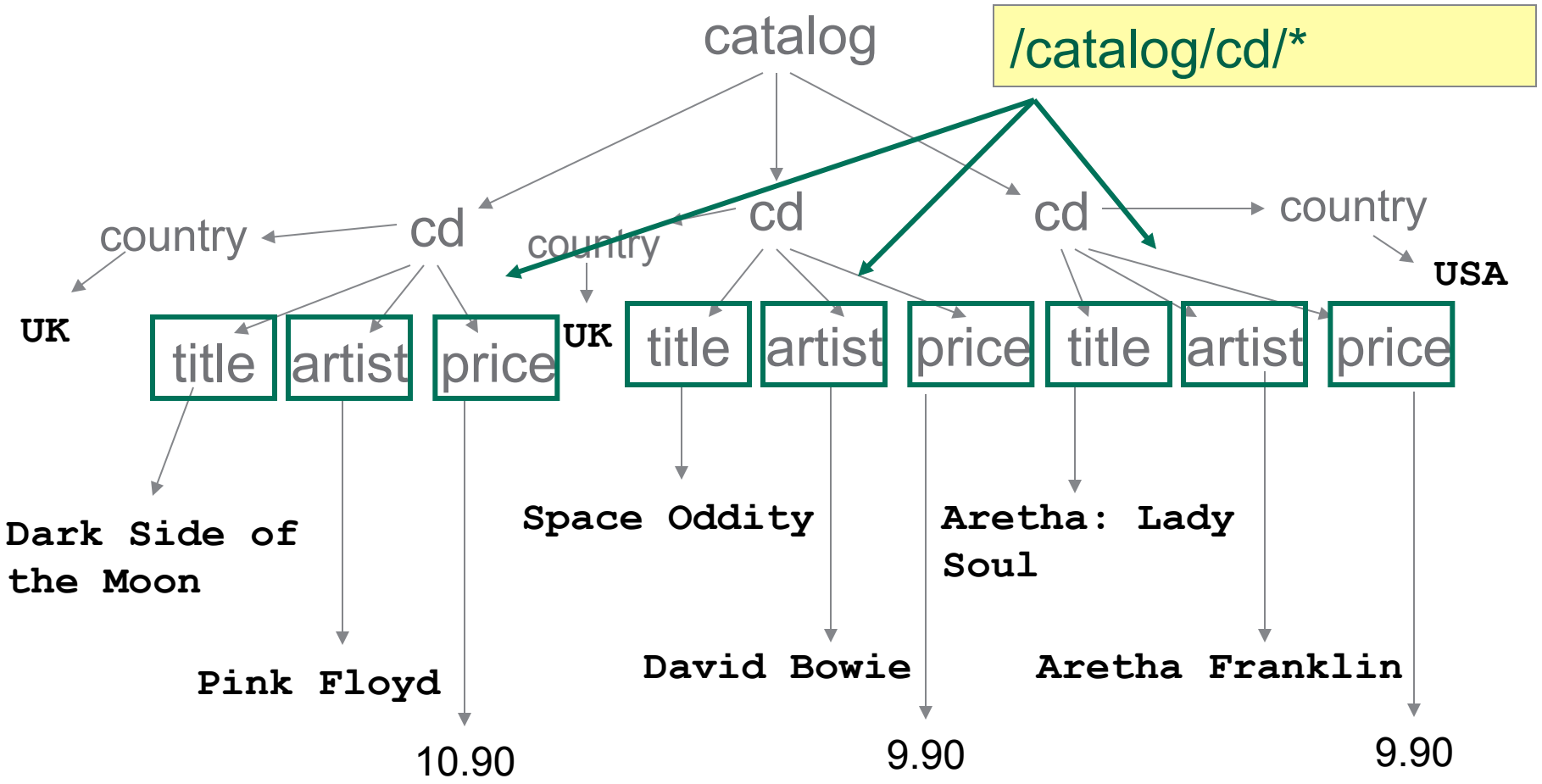


Condition on elements



// represents any top down path in the document

catalog.xml



* represents any element name in the document

What do the following expressions return?

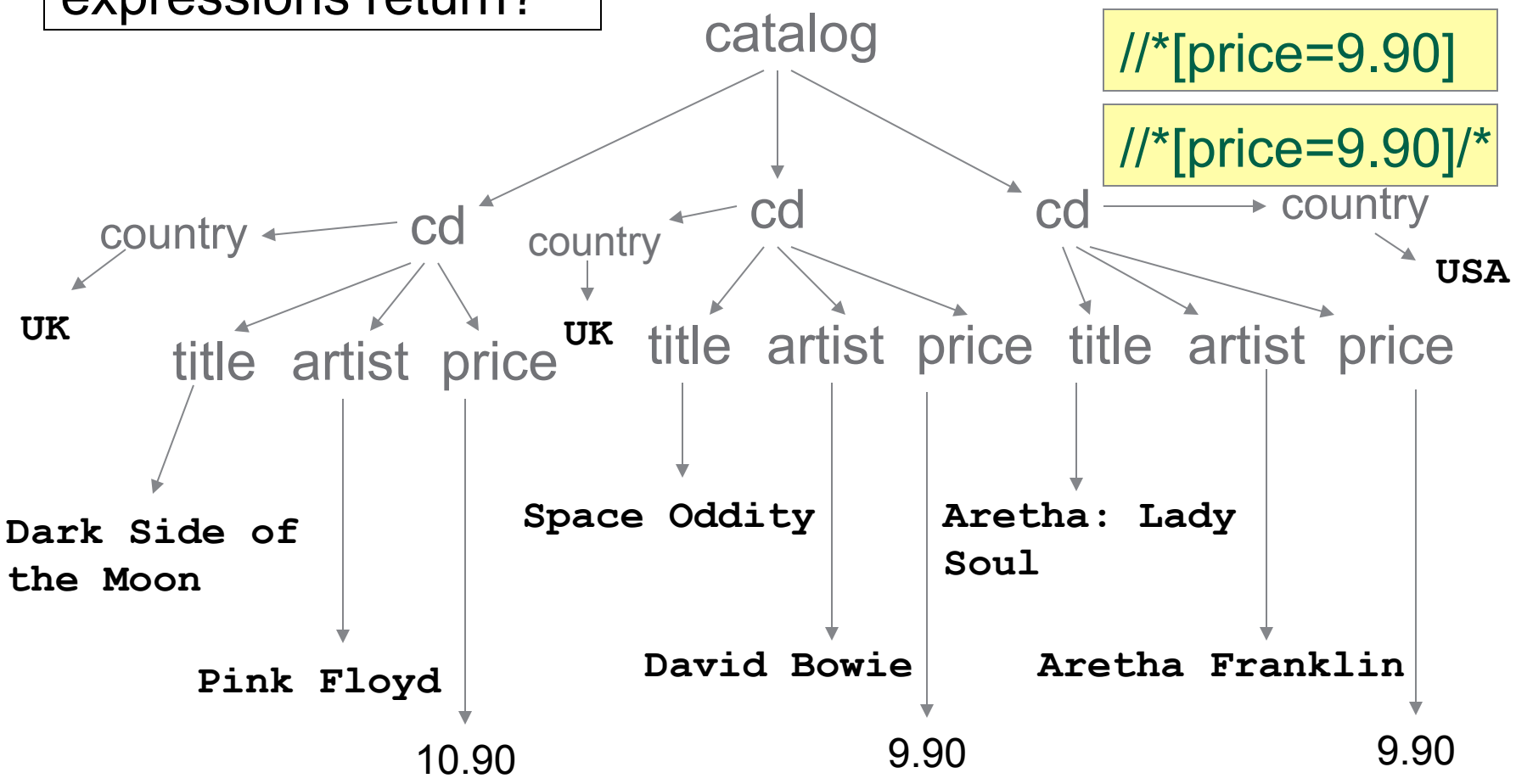
catalog.xml

/*/*

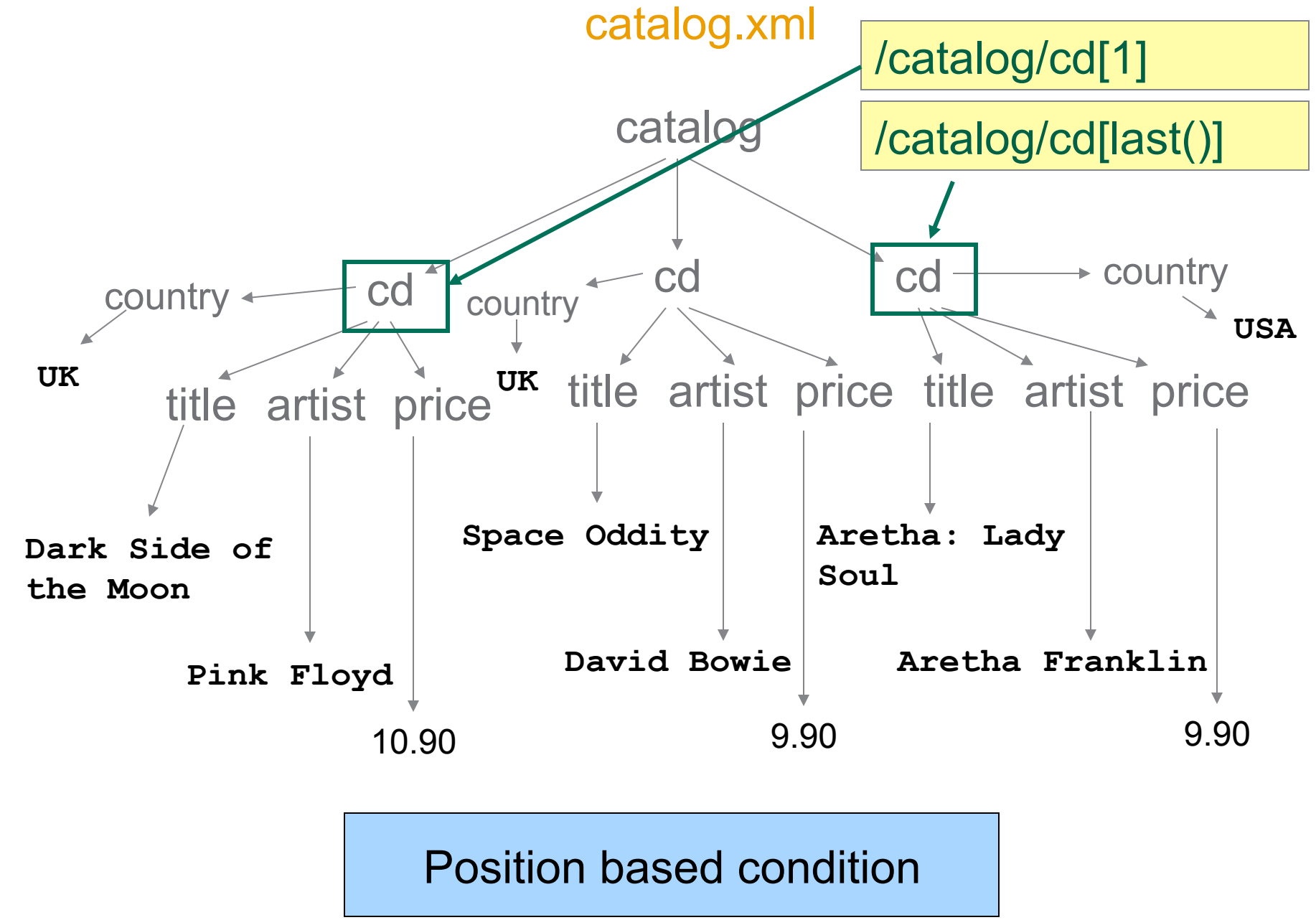
//*

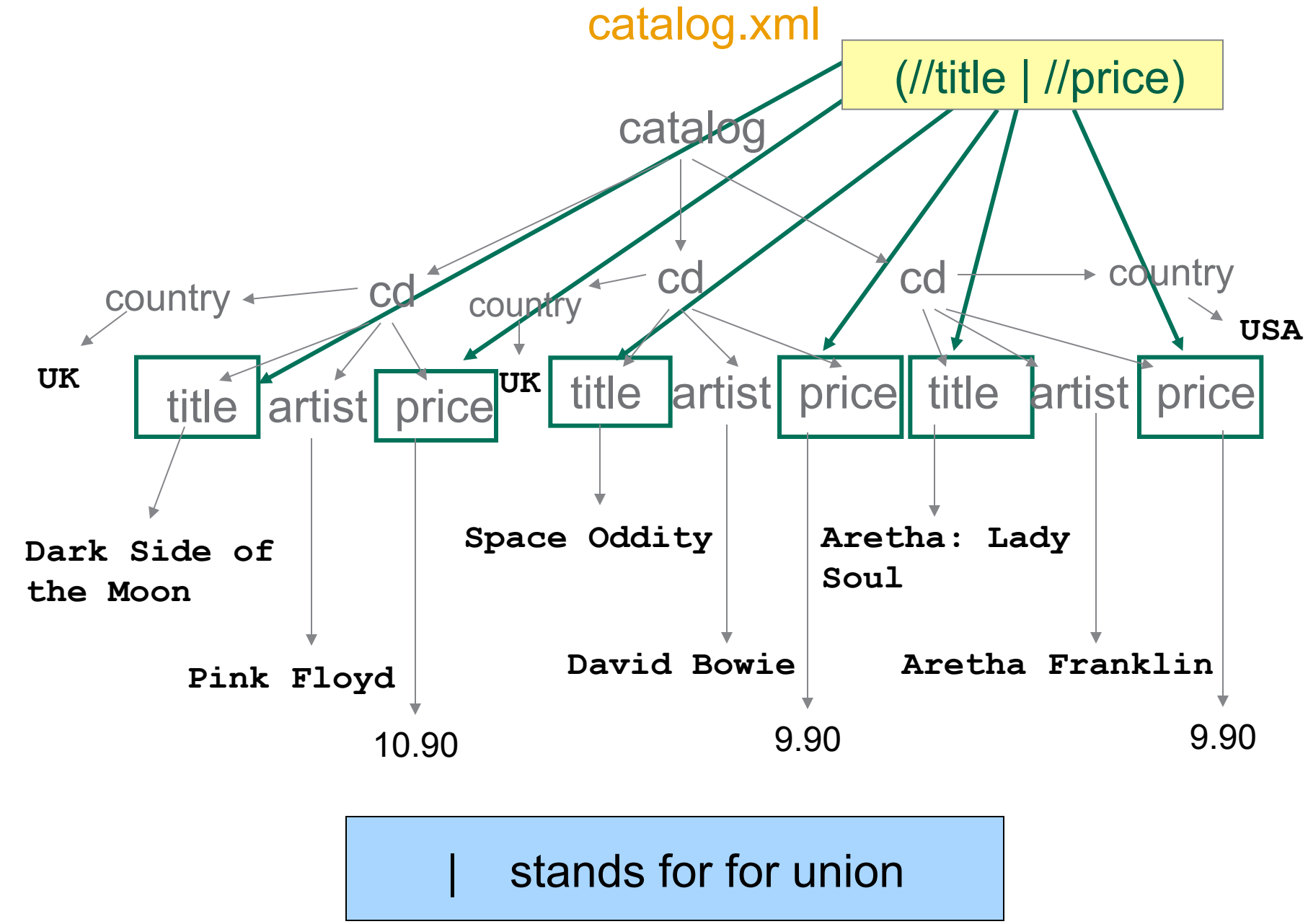
//*[price=9.90]

//*[price=9.90]/*



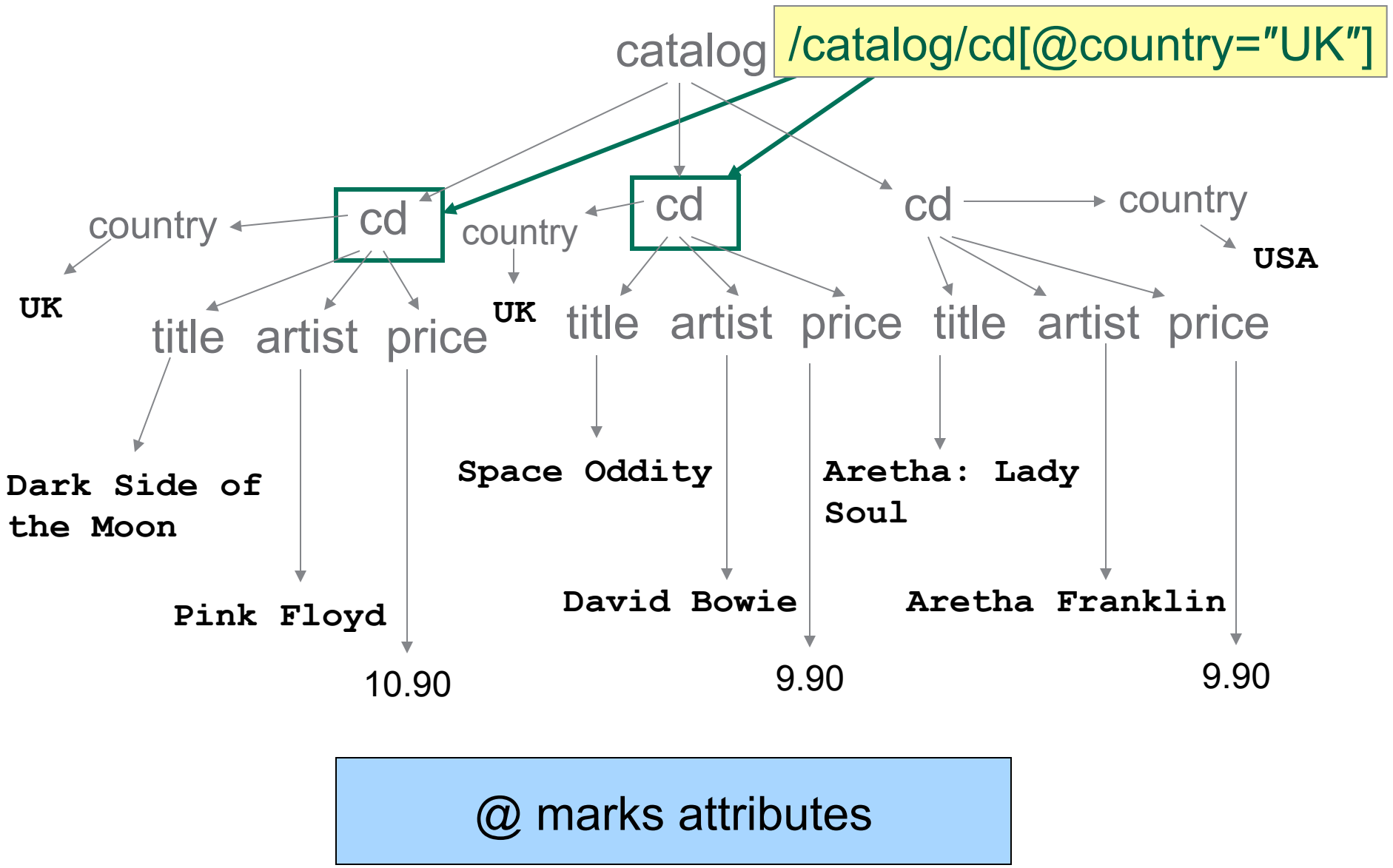
* represents any element name in the document


catalog.xml
/catalog/cd[1]
/catalog/cd[last()]
catalog
cd
cd
cd
country
country
country
UK
UK
USA
title
artist
price
title
artist
price
title
artist
price
Dark Side of the Moon
Pink Floyd
10.90
Space Oddity
David Bowie
9.90
Aretha: Lady Soul
Aretha Franklin
9.90
Position based condition



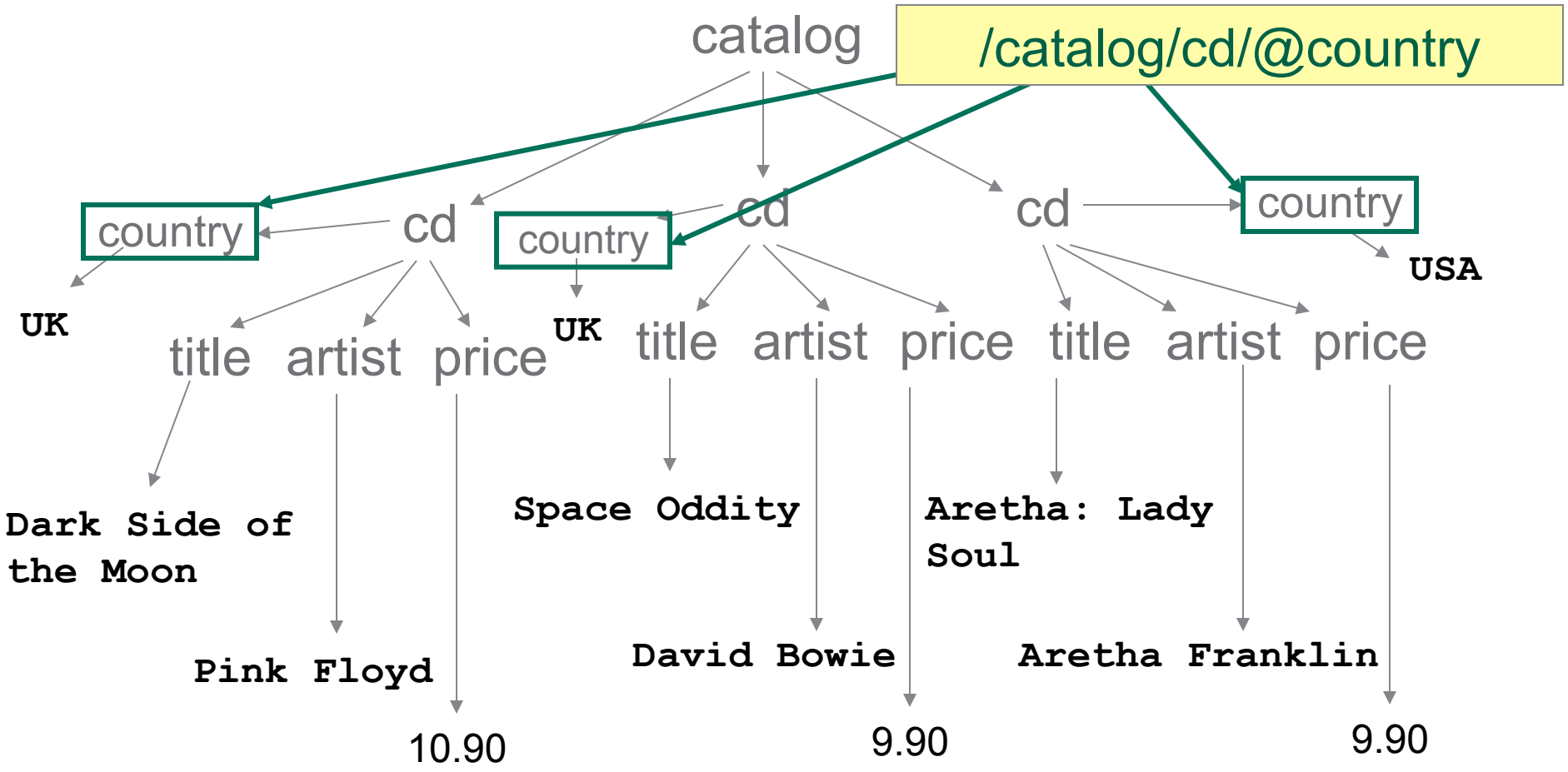
| stands for for union

catalog.xml



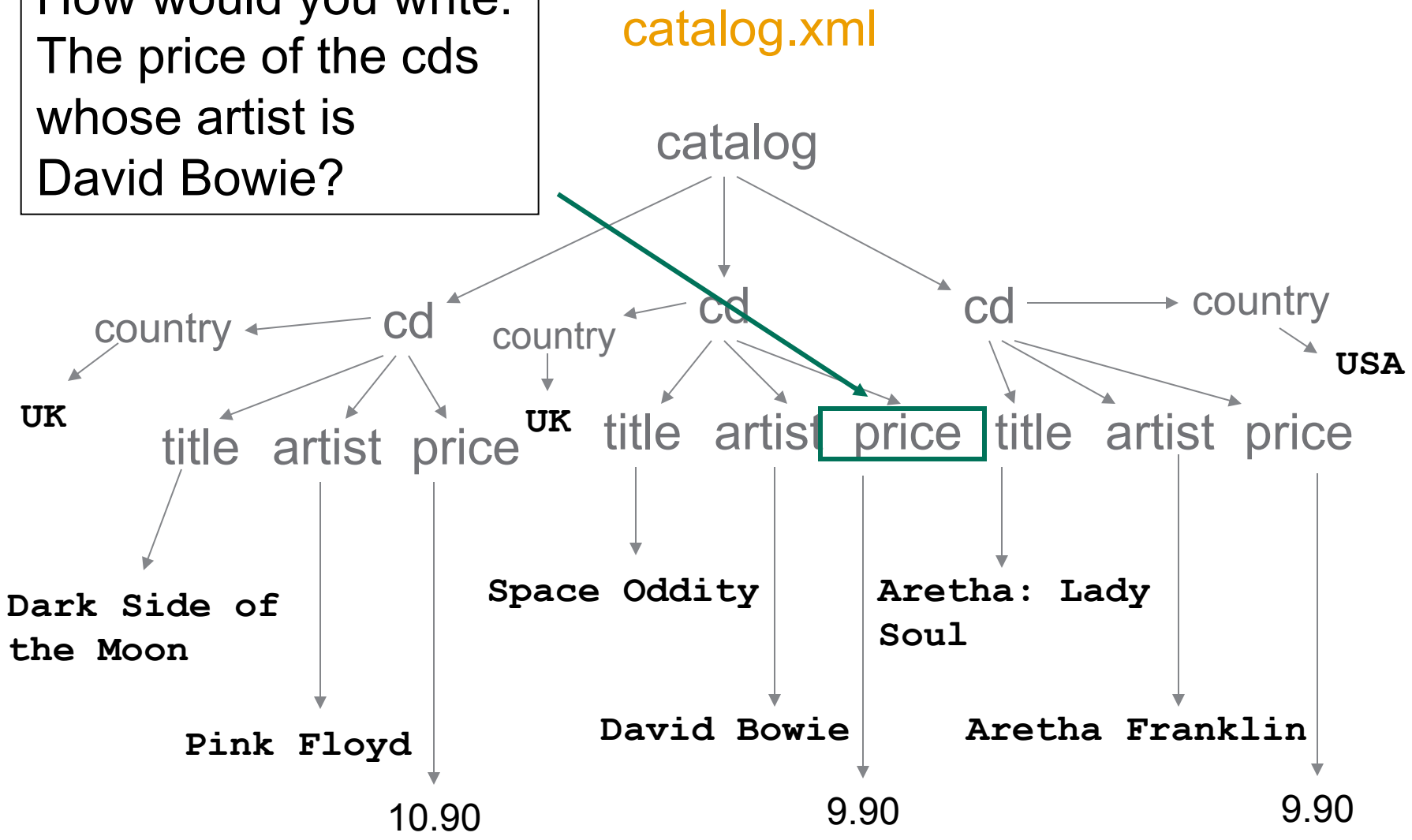
@ marks attributes

catalog.xml



@ marks attributes

How would you write: The price of the cds whose artist is David Bowie?



Navigational Axes *(plural of "axis")*

- We have discussed the following axes:
  - child (/)
  - descendant (//)
  - attribute (@)
- These symbols are actually shorthands, e.g.,

  /cd//price is the same as

  child::cd/descendant::price
- There are additional shorthands, e.g.,
  - self (/.)
  - parent (/..)

Additional Axes

ancestor	Contains all ancestors (parent, grandparent, etc.) of the current node
ancestor-or-self	Contains the current node plus all its ancestors (parent, grandparent, etc.)
descendant-or-self	Contains the current node plus all its descendants (children, grandchildren, etc.)
following	Contains everything in the document after the closing tag of the current node
following-sibling	Contains all siblings after the current node
preceding	Contains everything in the document that is before the starting tag of the current node
preceding-sibling	Contains all siblings before the current node

Info and Tools

You will find more info in the next lecture and:

- XPath 1.0 specification at W3C
  (there is also XPath 2.0, which is not yet widely supported)
- XPath tutorial at W3Schools
- Mulberry XPath Quick Reference

Tools for our course

- XPath plugin for Eclipse
- Saxon XSLT and XQuery Processor
- Kernow front end for Saxon *(I'll let you know the code for unlocking it)*
- XMLQuire XML and XPath Editor and Visualizer