Voting Patterns

Tuesday, 9/15/09

Thursday, 9/17/09

Recipe for solving the problem

- Find out number of votes in 109th congress
- Create a large table, with rows indexed by senator, and columns indexed by votes
- Foreach vote
  - Open webpage for that vote
  - If it's on "Passage of a Bill"
    - Foreach senator
      - Record his/her vote in the appropriate row/column
- *Compare each senator's record with Kennedy's*
- Sort the senators by Ted-ness

Missing ingredients

- Compare each senator's record with Kennedy's
  - Discuss during next class meeting
- Sort by Ted-ness
  - Sorting 100 numbers is pretty easy

Generalization

- …
- Compare each senator's record with *Kennedy's*
- Sort the senators by Ted-ness

- We can generalize: not just *Kennedy*, but *Mikulski*, or *Boxer*, or *Kerry.*
  - Discuss during later class

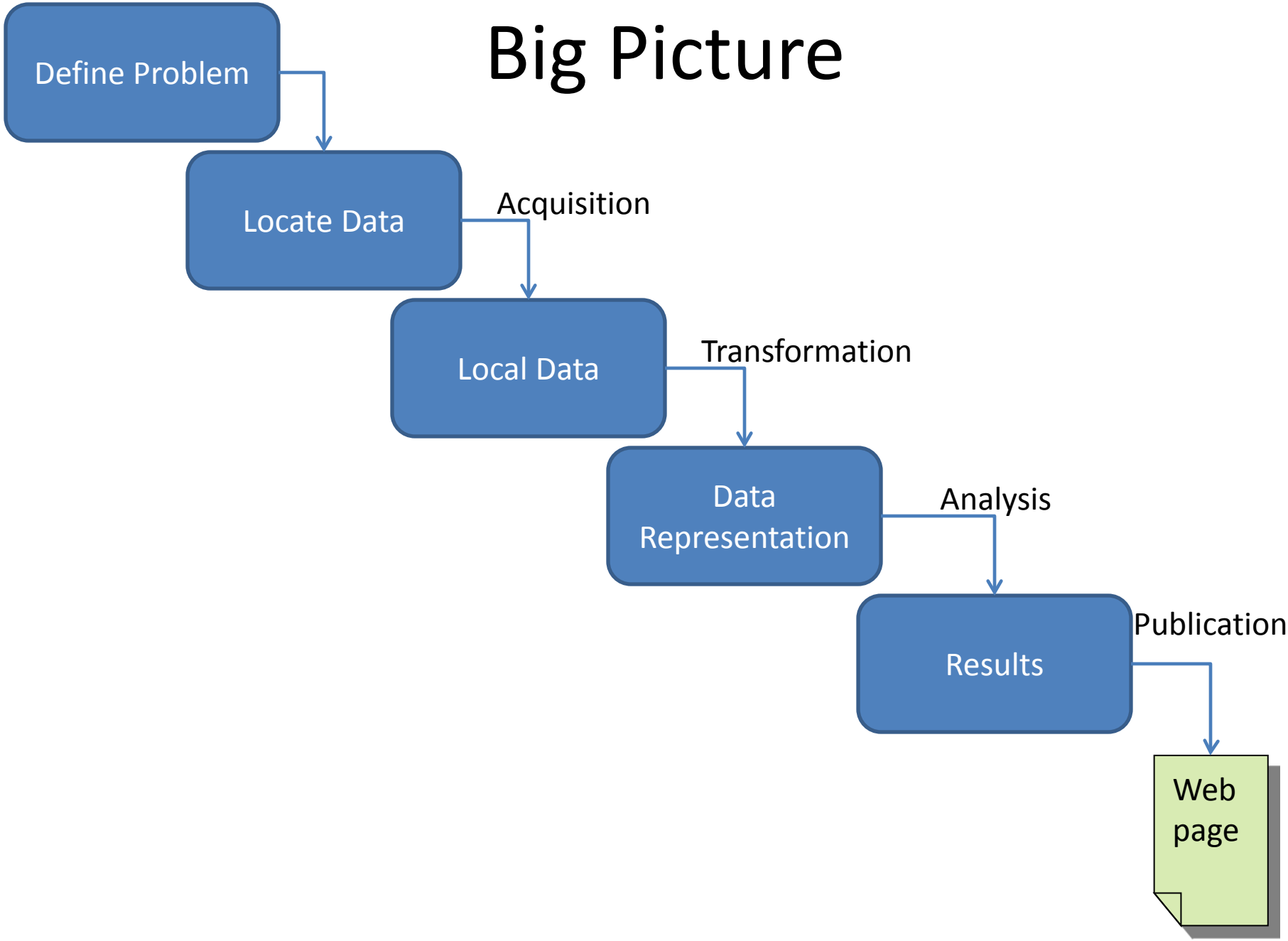
How long will it take?

Recipe for solving the problem

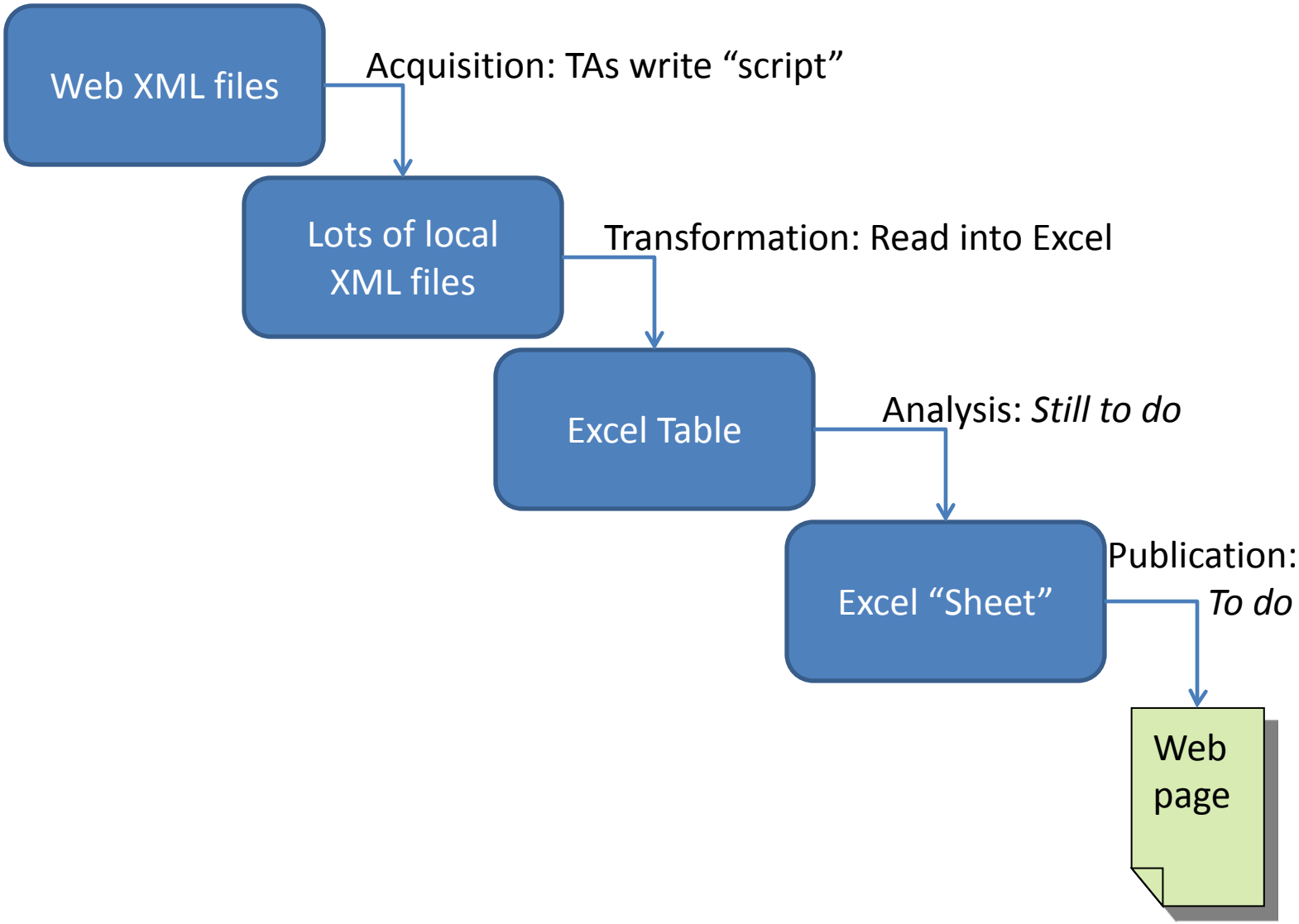
- Find out number of votes in 109th congress About 250
- Create a large table, with rows indexed by senator, and columns indexed by votes 101 x 250 table
- Foreach vote 250 times…
  - Open webpage for that vote …10 seconds
  - If it's on "Passage of a Bill" …3 seconds
    - Foreach senator 100 times…
      - Record his/her vote in the appropriate row/column 5 seconds

Work: 250 (13 secs) + 250 x 100 x 5 secs: about 86 **days** of work.

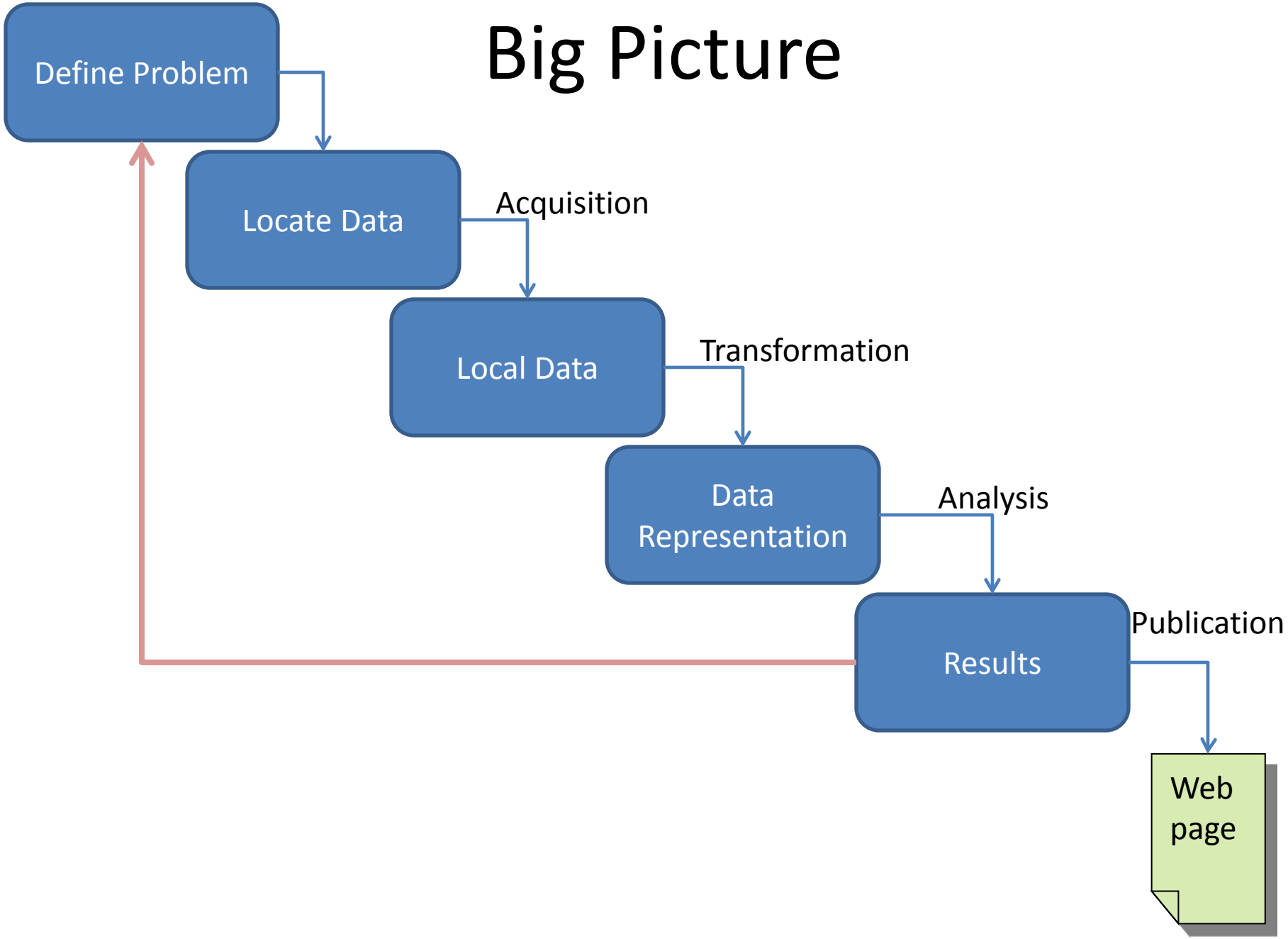
Big Picture


Define Problem
Locate Data
Acquisition
Local Data
Transformation
Data Representation
Analysis
Results
Publication
Web page

Senate Voting Example



Big Picture


Define Problem
Locate Data
Acquisition
Local Data
Transformation
Data Representation
Analysis
Results
Publication
Web page

Activity: look at XML page

XML Document Structure

- A computationally-motivated way to describe *documents*
- Very structured

```
<?xml version="1.0" encoding="UTF-8" ?>
<roll_call_vote>
        <congress>107</congress>
        <session>1</session>
        …
        <document>
                <document_type>H.R.</document_type>
                <document_number>333</document_number>
                …
        </document>
        <members>
                <member>
                        <member_full>Akaka (D-HI)</member_full>
                        <last_name>Akaka</last_name>
                        <party>D</party>
                        <vote_cast>Yea</vote_cast>
                        …
                </member>
                <member>
                        …
                </member>
        </members>
</roll_call_vote>
```

<?xml version="1.0" encoding="UTF-8" ?>

- *“We’re using XML; the character set we’re using is a really common one”*
- Stuff in pointy brackets <…> describes the document
- Stuff outside is the *content*

<roll_call_vote>

…

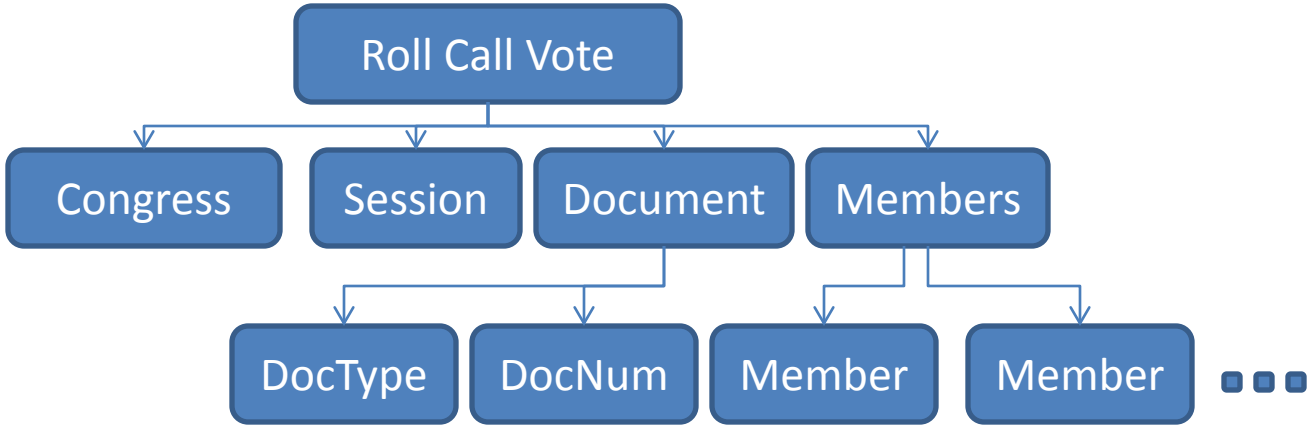
</roll_call_vote>

- Almost all brackets contain *tags*
- They come in matching pairs
- <foo> … </foo>
- Names are generally human-readable
- Names become column-labels in Excel!

<roll_call_vote>
  <congress>**107**</congress>
  <session>**1**</session>
  …
</roll_call_vote>

- Often open- and close-tag on same line
- Sometimes even a shorthand: <foo 15 /> means the same thing as <foo> 15 </foo>
- Indenting is optional…but it sure improves readability!

A different representation of a document



congr	sessic	document	document	member_full	vote_cast
107	1	H.R.	333	Akaka (D-HI)	Yea
107	1	H.R.	333	Allard (R-CO)	Yea
107	1	H.R.	333	Allen (R-VA)	Yea
107	1	H.R.	333	Baucus (D-MT)	Yea
107	1	H.R.	333	Bayh (D-IN)	Yea

```
<?xml version="1.0" encoding="UTF-8" ?
<roll_call_vote>
    <congress>107</congress>
    <session>1</session>
    …
    <document>
        <document_type>H.R.</do
        <document_number>333<
        …
    </document>
    <members>
        <member>
            <member_ful
            <vote_cast>Y
            …
        </member>
        <member>
            …
        </member>
    </members>
</roll_call_vote>
```

- The nesting of tags determines “parent/child” relationship in the tree

Why XML?

- XML is very general, and widely adopted
- One particular “flavor” of XML is “HTML”, in which many web pages were created
- Problems with HTML led to adoption of XML
- Very easy to make *program-readable* documents with XML
- Very easy to make programs *write* XML!
- Almost never written by humans