



Unlock Content™



# XQuery: Joining Content and Data

Mary Holstege  
Lead Engineer, Mark Logic  
WWW 2007 11 May 2007

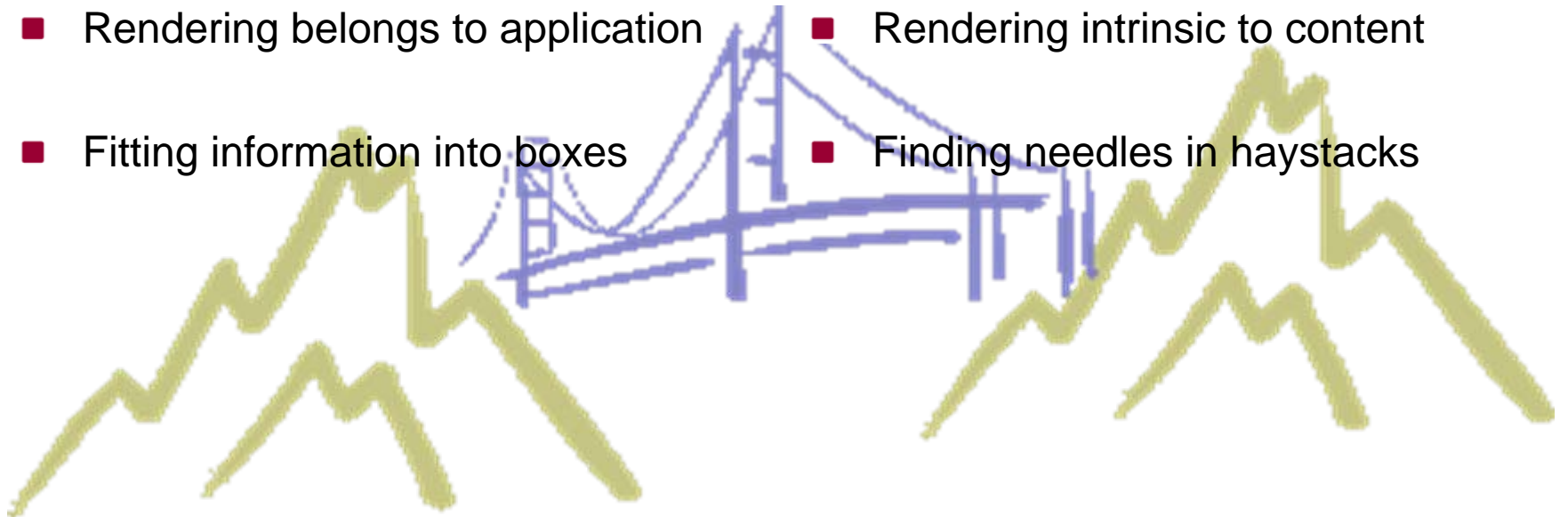
# Bridging the Data/Content Divide

## ***Data - RDBMS***

- Regular, uniform structure
- Defined rows and columns
- Strongly typed
- String values are simple
- Many small records
- High transaction volumes
- Rendering belongs to application
- Fitting information into boxes

## ***Content - Search***

- Irregular, varied structure
- Unknown or undefined structure
- Untyped
- String values may be compound
- Many large documents
- Few updates
- Rendering intrinsic to content
- Finding needles in haystacks



## ***Most Information Lives Somewhere in the Middle***

- Semi-regular structure
- Partially known structure
- Some strongly typed, some untyped
- String values often complex
- Mix of small and large documents
- Moderate levels of updates
- Rendering fluid
  
- Making sense of what you have



***It's a Gradient, Not a Divide***

- Content applications
  - Structure of data/content varied and varying over time
  - Structure may be unknown or incompletely known
  - Text for humans as well as atoms of data
  - Granular access plus information in context
  
- Playing with your content
  - Figuring out what you have: content discovery
  - Evolving and augmenting
  
- Split-brain syndrome

## *A great match!*

- Model general enough to fit variety of content
- Typed or untyped OK
- Easy to get started, scales to large applications
- Supports evolutionary development
- Works from back end to front end
- (XQuery Full-Text and Updates complete the picture)

## *How genealogy took over my living room*

- Basic data file
  - Strongly structured, fairly regular
    - But messy and full of estimates and uncertainties
  - But includes free-form notes and references to external sources
  
- Source data can be highly variable
  - Databases, sure
    - “Database” may be “gobs of OCRed and unprocessed records”
  - Archival material: letters, newspapers, land patents, photos
  
- Just an example to give a taste of how XQuery works for content applications

# Build It And They Will Come



- XML, sure
- SGML and HTML, fairly straight-forwardly
- Other textual formats, with a little work
- Non-textual formats, with conversion or metadata extraction
  
- Any data that can be made to look like an XML data model instance can play

# The Raw Data

0 @I00516@ INDI  
1 NAME Gerrit Jan Arie /Holstege/  
1 SEX M  
1 BIRT  
2 DATE 3 SEP 1891  
2 PLAC Ede, Netherlands  
2 SOUR @S258078@  
1 DEAT  
2 DATE 8 APR 1934  
2 PLAC Hillegersberg, Netherlands  
2 SOUR @S258078@  
1 BURI  
2 PLAC R.C. Cemetary, Enschede, Netherlands  
2 SOUR @S258078@  
1 OCCU  
2 PLAC Construction engineer  
2 SOUR @S258078@  
1 FAMS @F0222@  
1 FAMS @F0024@  
1 FAMC @F0223@  
0 @I00624@ INDI  
1 NAME Hendrikus Johannes /Holstege/  
1 SEX M  
1 BIRT  
2 DATE 19 JAN 1895

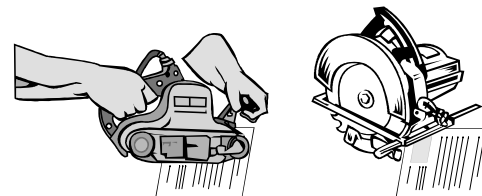


```
<INDI ID="I00516">
  <NAME>Gerrit Jan Arie /Holstege/</NAME>
  <SEX>M</SEX>
  <BIRT>
    <DATE>3 SEP 1891</DATE>
    <PLAC>Ede, Netherlands</PLAC>
    <SOUR REF="S258078"/>
  </BIRT>
  <DEAT>
    <DATE>8 APR 1934</DATE>
    <PLAC>Hillegersberg, Netherlands</PLAC>
    <SOUR REF="S258078"/>
  </DEAT>
  <BURI>
    <PLAC>R.C. Cemetary, Enschede, Netherlands</PLAC>
    <SOUR REF="S258078"/>
  </BURI>
  <OCCU>
    <PLAC>Construction engineer</PLAC>
    <SOUR REF="S258078"/>
  </OCCU>
  <FAMS REF="F0222"/>
  <FAMS REF="F0024"/>
  <FAMC REF="F0223"/>
</INDI>
```

```
<TITLE>Adressen Harderwijk 1908</TITLE>
</HEAD>
<BODY TEXT="#000000" LINK="#0000ff" VLINK="#551a8b" ALINK="#ff0000" BGCOLOR="#c0c0c0">
<P>update 6-12-2000</P>
<H4>Adressenlijst Harderwijk 1908,<BR>
met familienaam, voorletters, beroep, straat en huisnummer.<P>
Puntkomma gescheiden. Gesorteerd op achternaam en voorletters.
<P>De volgorde is:
<BR>voorletters;
<BR>achternaam;
<BR>beroep;
<BR>straat;
<BR>huisnummer<BR></H4>
<FONT FACE="Courier New" SIZE=3>
<P> P.J.T. van;Aarsen;sergt. ziekenopz. Mil. Hospitaal;;
<BR> A.;Aarts;koopman;Kromme Oosterwijk;322
<BR> P.;Aarts;fruithandel;Smeepoortstraat;19
<BR> Wed. J.H.J.;Aarts;inwonend;Wolleweverstraat;106
<BR> A.;Aartsen;mil. schoenmaker;Israelstraat;58
<BR> D.;Aartsen;kleermaker;Groote Poortstraat;294
<BR> F.;Aartsen;schoenmakersknecht;Heralttenstraat;48
...
```

# As Typed As You Wanna Be

- Completely untyped
- Completely strongly typed
- Some pieces typed
- Partially valid is completely OK
- Invalidity is not a capital offense



- Your content doesn't have to be perfect just to get started
- You can use all the power of XQuery to *make* it perfect

- Simple queries can accomplish a lot
  - Easy to get started

```
//FAM[@ID=//INDI[@ID="I00516"]/FAMS/@REF]
```
  - Exploring the variation in the data

```
//INDI[fn:count(BIRT) > 1]
//FAM>(* except (HUSB|WIFE|CHIL|MARR))
//DATE[fn:not(. castable as xs:date)]
```
- Layers of function libraries can build complete large-scale applications

```
view:person-to-xhtml(
  app:privatize-person(
    data:get-complete-person($name) ) )
```

# Scaling in the Data Dimension



- Functional language, with limited side-effects
  - XQuery Updates too
- Highly optimizable
  - Rewritable to take advantage of index, etc.
  - Lazy evaluation of large node sequences

# ▲ Simple Extraction and Display

- Direct translation to XHTML, CSS styling, links to navigate

...

```
<table>
```

```
<tr><th>Birth</th><td>{fn:data($person/BIRT/DATE)}</td>
```

```
<td>{fn:data($person/BIRT/PLAC)}</td>
```

```
<td>{let $ref := fn:data($person/BIRT/SOUR/@REF) return
```

```
<a href="get-source.xqy?id={$ref}">{$ref}</a>}</td></tr>
```

```
<tr><th>Death</th><td>{fn:data($person/DEAT/DATE)}</td>
```

```
<td>{fn:data($person/DEAT/PLAC)}</td>
```

```
<td>{let $ref := fn:data($person/DEAT/SOUR/@REF) return
```

```
<a href="get-source.xqy?id={$ref}">{$ref}</a>}</td></tr>
```

...

# Simple Extraction and Display

James C /Lindsey/ - Mozilla Firefox

File Edit View Go Bookmarks Yahoo! Tools Help

http://localhost:9001/simple1.xqy?id=100020

## James C /Lindsey/ (M)

<b>Birth</b>	1837 Ohio	
<b>Death</b>	1910	
<b>Buried</b>	1910 Ethridge Cemetery, Lawrence Co., Tennessee	<a href="#">S287625</a>
<b>Occupation</b>	1880 Blacksmith	<a href="#">S002147</a>
<b>Occupation</b>	1870 Benton, Hocking, Ohio	<a href="#">S002101</a>
<b>Occupation</b>	1850 Benton, Hocking, Ohio	<a href="#">S002032</a>
<b>Occupation</b>	1860 Benton, Hocking, Ohio	<a href="#">S002078</a>
<b>Occupation</b>	1880 Benton, Hocking, Ohio	<a href="#">S002147</a>
<b>Military</b>	Company A, 73 Ohio Regt, Civil War	

## Spouses

[F0019](#)

## Parents

Done

# Join and Aggregate

James C /Lindsey/ - Mozilla Firefox

File Edit View Go Bookmarks Yahoo! Tools Help

http://localhost:9001/simple2.xqy?id=I00020

## James C /Lindsey/ (M)

**Birth** 1837 Ohio

**Death** 1910

**Buried** 1910 Ethridge Cemetery, Lawrence Co., Tennessee [S287625](#)

**Occupation** 1880 Blacksmith [S002147](#)

**Occupation** 1870 Benton, Hocking, Ohio [S002101](#)

**Occupation** 1850 Benton, Hocking, Ohio [S002032](#)

**Occupation** 1860 Benton, Hocking, Ohio [S002078](#)

**Occupation** 1880 Benton, Hocking, Ohio [S002147](#)

**Military** Company A, 73 Ohio Regt, Civil War

## Spouses

**Married** 30 NOV 1865 Hocking Co., Ohio

**Wife** [I00019](#)

**Husband** [I00020](#)

**Child** [I00812](#) Natural Natural

Done



- Co-evolution of content and applications
  - Can use complex queries to augment and enrich content
  - Which enables more complex queries
  - Which lead to more augmentation and enrichment of content
  
- The more you do, the more you think of doing




- Spit and polish
  - Displaying relevant information in context
  - Prettier rendering
  - AJAX interactivity
  
- Data normalization and parsing: introduce typed attributes
  - Split out subfields (“James Clay /Lindsey/, Jr.”)
  - Ordering (“Est 1856”, “23 Jan 1900-1901”)
  
- Quality of information
  - Annotate content with quality information
  - How good is that source? For that kind of fact?

**Census 1920 Diamond, Haskell, Oklahoma; Roll: T625\_1462; Page: 1A; Enumeration District: 30; Image: 317.**

Name:  Spell?

Place:  Spell?

Words:


 Census 1920 Diamond, Haskell, Oklahoma; Roll: T625\_1462; Page: 1A; Enumeration District: 30; Image: 317.

US census

Internet

## Referenced by

[James Clay /Lindsey/, Jr. \(M\)](#)<sup>[1]</sup> b. 20 SEP 1878<sup>[4]</sup> d. 29 MAY 1955<sup>[1]</sup>

[Elza G /Moses/ \(F\)](#)<sup>[1]</sup> b. 18 DEC 1884<sup>[1]</sup> d. 2 JAN 1967<sup>[1][3]</sup>

[Bessie Lucille /Lindsey/ \(F\)](#)<sup>[1]</sup> b. 9 MAR 1903<sup>[2]</sup> d. 15 JAN 1985

Done

# Code Snippet

```
...
<h2>Referenced by</h2>
<div class="block">{
if (fn:empty(//INDI[//SOUR/@REF=$source/@ID])) then () else
<table border="0">
{
  for $person in //INDI[//SOUR/@REF=$source/@ID]
  order by person/@NUMERIC_DATE
  return gen:format-person-ref($person)
}</table>
,
if (fn:empty(//FAM[//SOUR/@REF=$source/@ID])) then () else
<table border="0">{
  for $family in //FAM[//SOUR/@REF=$source/@ID]
  order by $family/@NUMERIC_DATE
  return <tr><th align="left" valign="top">{gen:format-marriage-ref($family)}</th><td
  valign="top"></td></tr>
}</table>
}</div>
...
```

# James C /Lindsey/

## Marriage Lindsey and Pettit : 30 NOV 1865 Hocking Co., Ohio

m. 30 NOV 1865

**Wife: Sarah Jane /Pettit/ (F)**<sup>[1]</sup>

**Child: William /Lindsey/ (M)**

**Child: Arthur /Lindsey/ (M)**

**Child: Samual T. /Lindsey/ (M)**<sup>[4]</sup>

**Child: Mary /Lindsey/ (F)**

**Child: James Clay /Lindsey/, Jr. (M)**<sup>[2]</sup>

*Hocking Co.,*

b. 25 NOV 1865

b. 1867<sup>[1]</sup>

b. 1870<sup>[1]</sup>

b. MAR 1872<sup>[4]</sup>

b. JAN 1876

b. 20 SEP 1878<sup>[3]</sup>

Name:  Spell?

Place:  Spell?

Words:

- Can
- Canaan
- Canad
- Canada
- Canadi
- Canadian
- CANADIEN
- Canady
- Canal
- Canarreos
- Canceled
- Cancelled
- cancer
- cand
- Candace

b. 6 APR 1888

UNKNOWN

(missing)

d. AFT 1930<sup>[4]</sup>

(missing)

d. 29 MAY 1955<sup>[2]</sup>

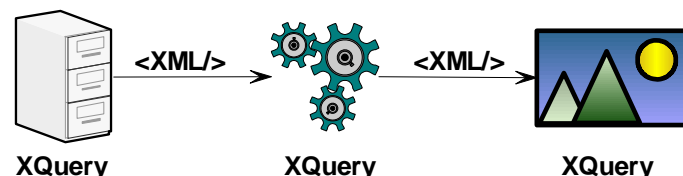
## Parents Lindsey and Beard : 10 OCT 1830 Muskingum, OH

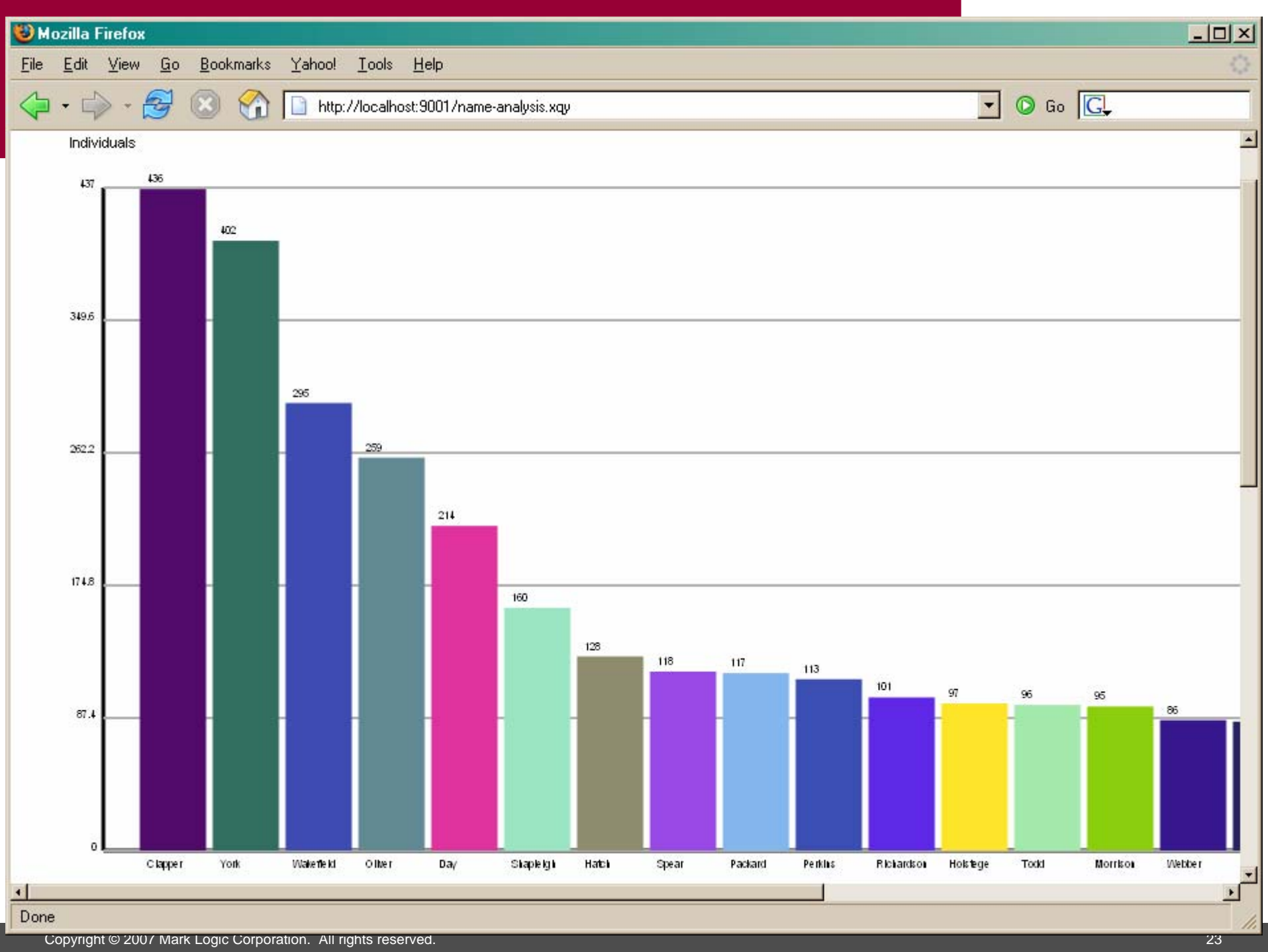
m. 10 OCT 1830

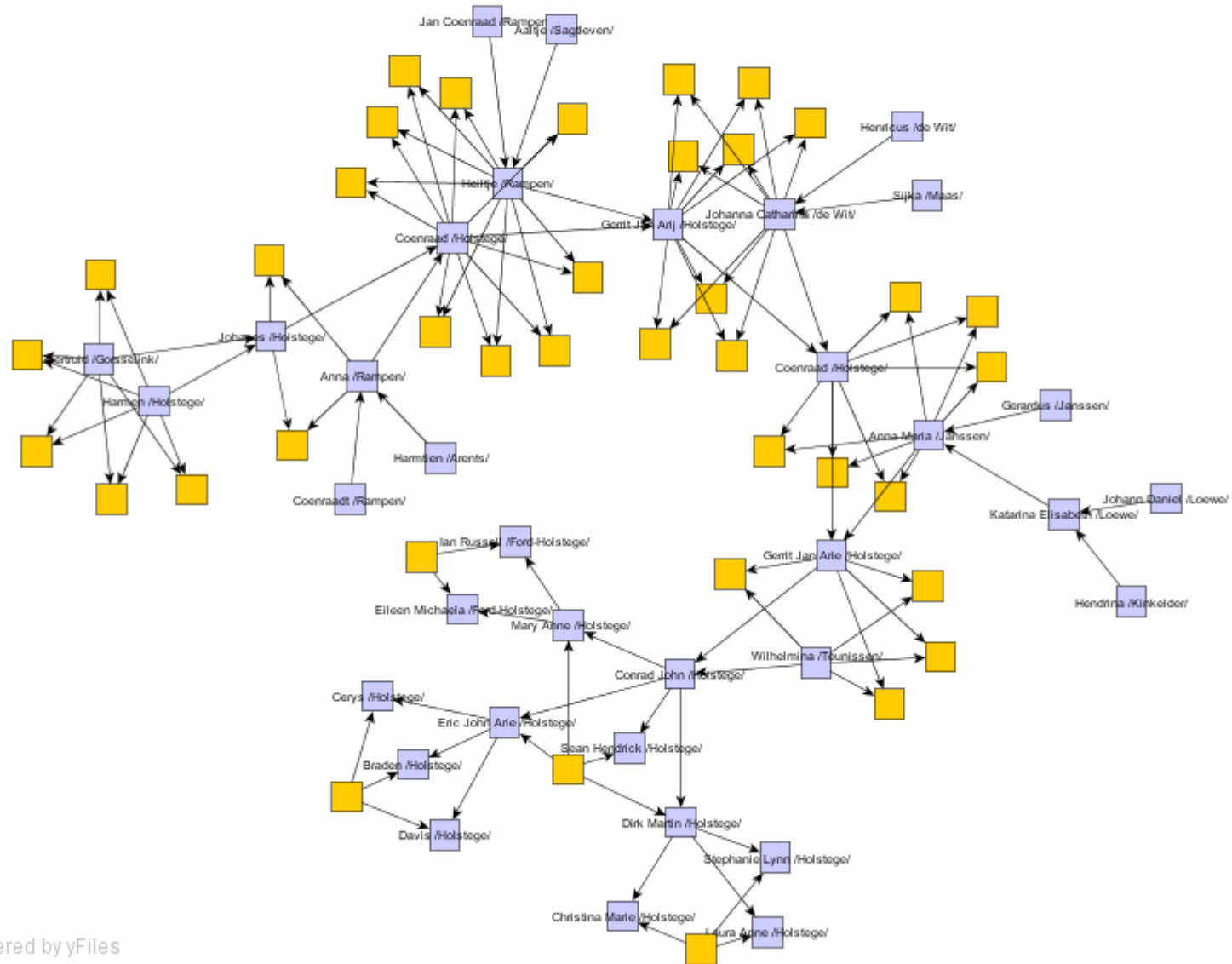
Muskingum, OH<sup>[6]</sup>

# XQuery Everywhere

- Data tier
  - Select, extract, aggregate
- Middle tier
  - Apply business logic to extracted data
  - Augment content
- Presentation tier
  - Render to browser (e.g. XHTML, SVG), other devices
  - Render for printing, sharing (e.g. XSL:FO, Office XML)
  - Export as other XML/textual formats
- Reduce friction between layers
- Rapid application development









```
declare function gen:people-to-graphml ($people as element(INDI)* ) as element(gr:graphml)
{
  <gr:graphml>
    <gr:key id="d0" for="node" yfiles.type="nodegraphics"/>
    <gr:graph edgedefault="directed"> {
      for $person in $people return
        <gr:node id="{ $person/@ID }">
          <gr:data key="d0">
            <y:ShapeNode>
              <y:NodeLabel visible="true" alignment="center">{ $person/NAME[1]/text() }</y:NodeLabel>
            </y:ShapeNode>
          </gr:data>
        </gr:node>
      ,
      for $fam in //FAM[ @ID=$people/(FAMS|FAMC)/@REF ] return (
        for $child in $fam/CHIL return (
          <gr:edge source="{ $fam/HUSB/@REF }" target="{ $child/@REF }"/> ,
          <gr:edge source="{ $fam/WIFE/@REF }" target="{ $child/@REF }"/>
        )
      )
    }</gr:graph>
  </gr:graphml>
}; (: people-to-graphml :)
```

- XQuery 1.0 provides the basics
  - Query, manipulate, render
- XQuery 1.0 Full Text extensions
  - Essential for human text, esp. multilingual text
- XQuery 1.0 Update extensions
  - Iterative improvement of content
  - Content annotation
- Extension function libraries for specific needs
  - e.g. HTTP GET/POST
  - e.g. security and access control
  - e.g. trigonometric functions

# Results for Atlanta

## Notes

[Author of one of our Civil War letters.1880 Census-Benton, Hocking Co., Ohio](#)  
[kelly anc.ftw]Littlefield Family Newsletter by Charles Littlefield Seaman, 1996, Vol 6 pg 65,66:  
[kelly anc.ftw]Littlefield Family Newsletter by Dorothy Keyes and Charles Littlefield Seaman, 1992, pg 78:

Name:  Spell?  
Place:  Spell?  
Words:

## Archival Material

### Letter dated 13 August 1864

This letter is postmarked January 4th in New York. The hand is very elegant, almost caligraphic. As indicated by the content, the regiment was still involved in the siege of Atlanta.

### 73rd Ohio Infantry

#### History

Organized December 30, 1861, under Colonel Orland Smith, it entered the field in January, 1862, operating in West Virginia until May, when it engaged Jackson in the Shenandoah Valley, and participated in the battle of Cross Keys. In August it took conspicuous part in the second battle of Bull Run, acting with great gallantry and losing about 150 men. It remained near Washington until December, when it joined Burnside at Fredericksburg, and in April, 1863, was at Chancellorsville. The Regiment moved north in June, and participated in the battle of Gettysburg in July, with a loss of 143 men. In September it was transferred with Hooker's command to the

## *Joining Data and Content*

- Query
  - Navigate content
    - > Typed or untyped
    - > Well-structured, inconsistent structure, unknown structure
  - Search text as text (XQuery Full-text)
    - > But with fine-grained, structural knowledge
  
- Manipulate
  - Annotate, enrich, refine content (XQuery Updates for persistence)
  - Process typed data in a type-aware way
  
- Render
  - Constructed views, slices, transliterations, mash-ups
  - XHTML+CSS, RSS, SVG, XSL:FO, Office XML, GraphML...

# Thank You

Mary Holstege

[mary.holstege@marklogic.com](mailto:mary.holstege@marklogic.com)



# More Complex Searches and Indices

- Alternative slices and views
  - Index of people by last name
  - Subtrees: ancestors of, descendants of
  - Reverse lookup: where is source used?
  - People alive in 1850 for which there is no residence information
- Full text searches
  - Search for names, locations, word
  - Thesauri
- Relationship searches
  - Find “Ann” and “Jack” in the same family
- Analytics
  - Counts by name, by location



# Basic Statistics

- A week of work
- 17 MB GEDML + 155 KB thesaurus data + 30 MB text archives
- 400 MB source data (image) => 20KB metadata
- 2000 lines of XQuery
- 300 lines of JavaScript
- 150 lines of CSS
- (1500 lines of Java for original data conversion)