Saxon CE
Interactive browser based applications without a single line of Javascript

Jirka Kosek

http://www.kosek.cz
http://xmlguru.cz
jirka@kosek.cz
@jirkakosek

Slides at http://www.kosek.cz/xml/2013webexpo
XSLT Introduction

- Language for transforming XML to HTML, other kinds of XML or plain-text
- Version 1.0 from 1999 is supported by all browsers
- Version 2.0 from 2007 provides many new features
- Ultimate templating language
- Operates on document tree not on text stream
XSLT Example

<employee>
  <name>John</name>
  <salary>80000</salary>
</employee>

<xsl:template match="employee">
  <tr>
    <td><xsl:value-of select="name"/></td>
    <td><xsl:value-of select="format-number(salary, '#,##0.00 USD')"/></td>
  </tr>
</xsl:template>

| John   | 80,000 USD | Output |
Why average Web developer hates XML?
Because of DOM

Documented Obscure Model

Documented Object Mess

Document-Object-Model

Death Or Madness

... fill in your favorite one ...
Why good developer should not abandon XML

• JSON is cool now
  ◦ but it makes sense mainly for communication between browser and backend

• XML is deployed everywhere from enterprise systems to your media center

• Better APIs should be used
  ◦ where is my E4X?

• Higher level query and transformation languages should be used
  ◦ XPath, XQuery, XSLT, JSONiq, …
What's bad with the current XSLT support in browsers

- Only old version 1.0 is supported
- Only classical static XSLT approach is supported
  1. Load XML
  2. Transform it to HTML
  3. Render resulting HTML
- No interactivity
- How to get XSLT 2.0 and interactivity into browser?
- Browsers are fast now, so complete and fast XSLT engine can be written entirely in Javascript
Saxon CE (Client Edition)

- Saxon is Java/.NET leading open-source and commercial XSLT/XQuery engine
- Saxon CE (Client Edition) is a streamlined version compiled into Javascript using GWT
- Provides XSLT 2.0 support in recent browsers
- Extends XSLT 2.0 by features allowing writing fully interactive pages
  - template can handle event
  - access to DOM and Javascript objects
  - read/write DOM access using XSLT features
- Transformation is not run at once, but can react to events and modify DOM
- Of course, Saxon CE be can be also used only as a replacement of built in XSLT 1.0 engine
Demo

- Event handling

```xml
<xsl:template match="button[@id = 'pressme']" mode="ixsl:onclick">
  ... handle event ... 
</xsl:template>
```
Demo

- DOM manipulation

```xml
<xsl:result-document href="#result"
  method="ixsl:replace-content">
  <table>
    <tr>
      <td>Foo</td>
      <td>Bar</td>
    </tr>
  </table>
</xsl:result-document>
```
So should I abandon jQuery and use Saxon CE instead?

- No

When is Saxon CE/XSLT 2.0 suitable:
  - You need to process and render complex XML in browser
  - Rendering logic is shared between server and client
  - Longer first load time is not a problem (Intranet applications)
Conclusion

- Give it a try and download from http://www.saxonica.com/ce/index.xml
- Attend XML Prague conference to learn more about recent XML developments http://xmlprague.cz

Any questions?